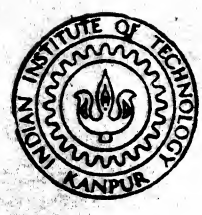
MODELLING OF HIGH TEMPERATURE PLASTIC DEFORMATION OF TWO PHASE STRUCTURES USING FINITE ELEMENT METHOD

By

SUDIPTO GHOSH



DEPARTMENT OF METALLURGICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

DECEMBER , 1991

MODELLING OF HIGH TEMPERATURE PLASTIC DEFORMATION OF TWO PHASE STRUCTURES USING FINITE ELEMENT METHOD

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY

By

SUDIPTO GHOSH

to the

DEPARTMENT OF METALLURGICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

DECEMBER, 1991

DEDICATED
TO

MA
and
BABA

AND

MAMIMA
and
MEJOMAMA

CERTIFICATE

This is to certify that the work entitled, **"Modelling of High Temperature Plastic Deformation of Two Phase Structures Using Finite Element Method"** has been carried out by **Mr. Sudipto Ghosh** under our supervision and that it has not been submitted elsewhere for a degree.

(**S. Bhargava**)
Assistant Professor

(**G. S. Murty**)
Professor

Department of Metallurgical Engineering
Indian Institute of Technology
Kanpur

ACKNOWLEDGEMENT

I take this oppourtunity to express my deepest sense of appreciation and gratitude to my supervisors Prof.G.S. Murty and Dr.S. Bhargava for the sincere guidance, valuable suggestions and affection they have given during the course of my project work.

The valuable help and encouragement in getting grasp over Finite Element Method and its programming for the present investigation from Dr.T. Sundararajan cannot be expressed in words.

The valuable help extended to me by Mr.S.K. Sinha, Mr.D. Sanyal, Mr.Shamsi and Mr. Deshpande is also duly acknowledged.

I am grateful to S.K. Choudhary, Sommonnoy, Kailash, Amit, Praveen, Gour, Tapas, Pravirda and S. Sagar for the love rendered to me during the course of this study.

I have the oppourtunity to express my deep regards to Chandra uncle for his love and encouragement in the course of my work.

Love and affection received from Saritadidi, Subrat and Akun is duly acknowledged.

I cannot express in words the love and hapiness I have received with the company of Japesh, Subir, Sunil, Chandanda, Ajit, Dinesh, Potty, Minda, Nilesh, Alok, Akshay, Arul, Venky, Sajal, Rajesh, Sudipto, Anirvan and Amit during the course of this study.

The timely help of Mr.S. Singh, Mr.S.G. Chowdhury, Mr.K.K. Singh and Sharma uncle is duly acknowledged.

The days and nights of constant help during the processing of the thesis by Mamuji, Tapas and Shivani can never be repayed by anything other than love.I am also grateful to Mr. B.K.Jain for tracing the figures.

The sincerity, love, patience and cooperation rendered to me by my family members, Ma, Baba, Jethimoni, Jethu, Mamuji, Mamima, Dipan, Bubu and Buban during the course of my work cannot be repayed. Light moments with Buban (Bhompu) is to be mentioned specially. The encouragement of Dinanath, who is my constant companion since my childhood and love of Arindam during my B.Tech days is duly acknowledged.

Let me pay my humble obeisances to all the spirtual masters of the world. I am unable to express in words my supreme regards to Srila A.C. Bhaktivedanta Prabhupada and Sri Ramakrishna Paramhansa Deva whose direct words have attracted me towards Gauranga Mahaprabhu the most kind and unique incarnation of all attractive Krishna, the Supreme personality of Godhead. I owe to the same for happiness and peace during the course of my work.

CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

A_e	2-D element domain
a_i	weightages of gauss points
C	contiguity
D	physical domain
D_e	element domain
I_1	first invariant
I_2	second invariant
I_3	third invariant
J	jacobian
det[J]	determinant of jacobian
L/D	aspect ratio of plate
M_i	linear shape function
m	plate orientation
m'	strain rate sensitivity
N_i	quadratic shape function
P	pressure
R	residue
t_x	x-component of traction
t_y	y-component of traction
U	velocity component in x-direction
V	velocity component in y-direction
V_f	volume fraction
W_i	weightage function
Γ	boundary of the physical domain
Γ^e	boundary of element
δ_{ij}	kronecker delta

δ_x	displacement in x-direction
δ_y	displacement in y-direction
δ_z	displacement in z-direction
ε_{ij}	components of strain tensor
$\dot{\varepsilon}_{ij}$	components of strain rate tensor
ε'_{ij}	deviatoric strain tensor components
$\dot{\varepsilon}'_{ij}$	deviatoric strain rate tensor components
η	local y-coordinate
μ	pseudo viscosity
ξ	local x-coordinate
ρ	density
$\bar{\sigma}$	flow stress
$\underline{\underline{\sigma}}$	stress tensor
σ_{ij}	components of stress tensor
σ'_{ij}	components of deviatoric stress tensor
σ_y	yield stress
ϕ	exact solution
ϕ_i	nodal values
ϕ^*	approximate solution
$\vec{\nabla}$	gradient operator

CHAPTER 1

INTRODUCTION

Most of the commercial alloys used for several structural components have two-phase or multi-phase structures. Examples of such alloys are plain carbon and alloy steels, nickel base super alloys, brasses, titanium alloys and aluminium alloys. Among steels the specific examples are spheroidized steels (ferrite-cementite), dual phase steels (ferrite-martensite) and duplex stainless steels (austenite-ferrite). Alpha+beta titanium alloys such as Ti-6Al-4V are examples of alloys having B.C.C. β phase and H.C.P. α phase. Many advanced materials which are currently being developed, like intermetallics and metal matrix composites, also have two-phase/multi-phase structures. The presence of more than one phase often enables the development of structures which are efficient in load carrying applications and their properties can be varied over a wide range [1].

1.1 Mechanical Properties of Two-phase Alloy

Mechanical properties like yield-strength[1-3], hardness[1], ductility[3], creep[4], fatigue[3], fracture mode[3], and the coefficients of constitutive equation describing plastic flow[5,6] in a given alloy have been found to vary significantly with the distribution and volume fraction of phases present. This dependence is over and above their variation with grain size, sub-grain size, texture, coherency strain between different phases and interface phase, if present.

Fig.(1.1) shows the typical variation of yield strength versus the volume fraction of second phase in two phase alloys[1]. Ankem and Margolin[2] have given the following expression for the yield strength of a two phase alloy.

$$\sigma_Y = \sigma^C_{Y\alpha} V_\alpha + \sigma^C_{Y\beta} V_\beta + I^Y_{\alpha\beta} \qquad ...(1.1)$$

where V_α and V_β are volume fractions of α and β phase, $\sigma^C_{Y\alpha}$ and $\sigma^C_{Y\beta}$ are their bulk yield strengths and the term $I^Y_{\alpha\beta}$ is due to the interaction between the phases. This expression shows that the law of mixture for the yield strength need not be followed. The yield strength of the alloy not only depends on the volume fractions of two phases, V_α and V_β, but also on the interaction parameter $I^Y_{\alpha\beta}$ which is likely to depend on the morphology of two phases.

Similarly, hardness for two-phase alloys has been shown to vary as[1],

$$H = H_\alpha(1 - CV_\beta) + H_\beta(CV_\beta) \qquad ...(1.2)$$

where H_α and H_β are hardnesses of α and β phases, V_β is the volume fraction of β-phase in the alloy and C is termed as contiguity.

Fig.(1.2) shows true-stress vs true-strain curves for four $\alpha+\beta$ Ti alloys with different volume fraction of α-phase[5]. Significant rise in stress-value at a given value of strain (much above yield strain) with volume fraction of α phase can be observed. Sastry et al[6] observed that flow stress as a function of strain-rate, significantly varies with volume fraction of α-phase.

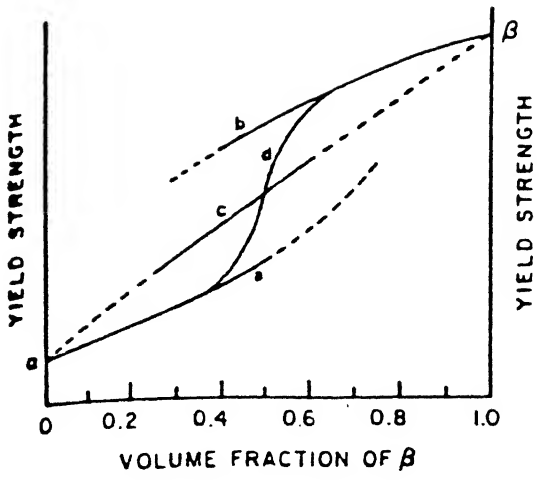


Fig.1.1 Typical curves showing various forms of the variation of yield strength versus volume fraction β of two-phase alloys

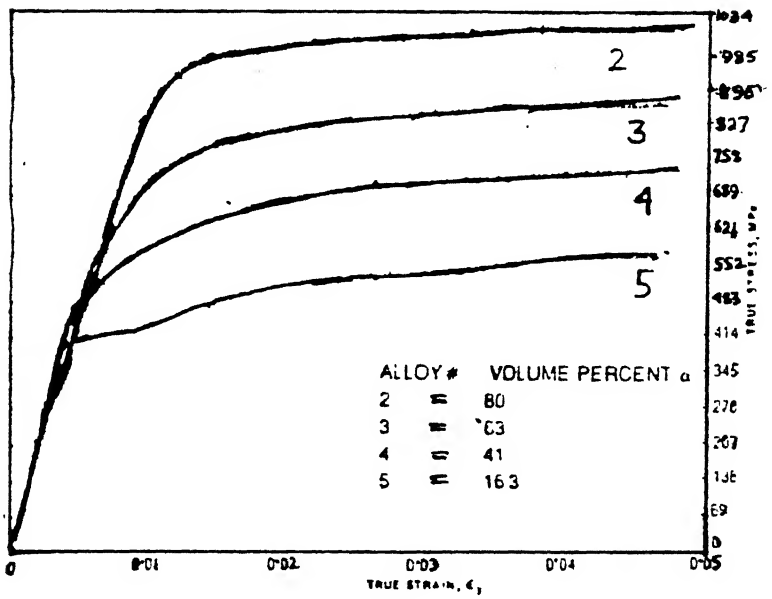


Fig.1.2 True stress - true strain curves for four (α+β) Ti alloys having fine α-phase particle and different volume fraction of α-phase

A significant amount of work has been done on the variation of properties of titanium alloys with phase morphology. Table(1.1) shows the variation in tensile properties of Ti-6Al-4V alloy with different thermo-mechanical treatments i.e. with different sizes and morphology of α-plates. It can be seen that both yield strength and ductility can be varied significantly. Similarly, Table (1.2) shows that fatigue properties also vary substantially with α-plate morphology. Creep strength is also sensitive to α-plate morphology in Ti-alloys and is the basis of evolving modified heat treatments in several high-temperature titanium alloys. Table (1.3) shows the variation of toughness in Ti-6Al-4V alloy with α-plate morphology. Microstructural features thus play a vital role in mechanical properties of two-phase/multi-phase alloys.

1.2 Objectives of Designing a Deformation Processing Schedule for Two-phase Alloys

One of the most important challenges faced by the design engineers and manufacturers of various commercial alloys is obtaining the desired microstructure in the as processed material so that the desired property levels in them can be obtained. This is conventionally done by utilizing the heat-treatment principles. However, the recent increase in emphasis on damage-tolerant design[3] has led to the realization that microstructure must also be controlled on a scale which is not significantly affected by heat treatment. An effective way of manipulating microstructure on such a scale is through combined

TABLE 1.1

Tensile Properties of Three $\alpha+\beta$ Ti-Alloy (Ti-6Al-4V)[3]

Sl. No.	Morphology of α-phase	Condition	Yield strength (MPa)	Tensile strength (MPa)	Elongation (%)	Reduction in area (%)
1	equiaxed	$\alpha+\beta$ forge + recrystallization anneal	711	876	12.4	36
2	equiaxed (coarse)	$\alpha+\beta$ forge + mill anneal (minm.values)	828	897	10.0	25
3	equiaxed + plate like	$\alpha+\beta$ forge + STA (age 4h at 594°C)	876	938	15.2	34
4	equiaxed + plate like	$\alpha+\beta$ forge + STOA (age 24h at 594°C)	904	973	15.5	47
5	plate like	β forge + AC (BA) (+705°C/ 2h/AC)	773	856	11.2	23
6	plate like	β forge + WQ (BQ) (+705°C/ 2h/AC)	863	932	5.9	6
7	equiaxed	$\alpha+\beta$ forge DA (870°C/2h/AC +705°C/2h/AC)	856	911	15.3	47

TABLE 1.2

Fatigue and Tensile Data for Various Microstructural Conditions of $\alpha+\beta$ Ti-Alloy (Ti-6Al-4V)[3]

Sl. No.	Condition and α-phase morphology	Yield strength (MPa)	Tensile strength (MPa)	Elongation (%)	Stress (smooth) at 10^7 cycles, (MPa)	Stress (notched) at 10^7 cycles, (MPa)
1	10% equiaxed primary α + anneal	971	1068	14	537	214
2	40% equiaxed primary α + anneal	930	1013	15	579	225
3	10% equiaxed primary α + STOA	978	1061	15	489	220
4	50% elongated primary α + anneal	923	1020	13	620	227
5	β forge + anneal (plate like)	882	992	11	565	220
6	β forge + STOA (plate like)	978	1075	10	586	220
7	10% equiaxed primary α + anneal	882	985	13	620	214

TABLE 1.3

Fracture - Toughness Variations in $\alpha+\beta$ Ti-Alloy (Ti-6Al-4V) for Plate Shaped Specimen[3]

Sl. No.	Morphhology of α-phase	Condition	Yield strength (MPa)	Tensile strength (MPa)	Elongation (%)	K_{IC} (K_Q) MPa$\sqrt{m}$
1	equiaxed	$\alpha+\beta$ roll + mill anneal	1096	1171	14	32
2	equiaxed	$\alpha+\beta$ roll + recrystallization (RA)	1054	1144	13	51
3	equiaxed plate like	$\alpha+\beta$ roll + duplex anneal (954°C/1.5h/AC+760°C/1h/AC)	882	971	14	59
4	plate like	BA ($\alpha+\beta$ roll +1038°C/20min/AC +732°C/2h/AC)	875	951	11	87
5	equiaxed	$\alpha+\beta$ roll + recrystallization anneal (RA)	785	882	-	94

thermal and mechanical working. Such a processing schedule allows a broader range of microstructures which can be obtained than that is possible through thermal treatment alone. Such manipulations are generally known as thermo mechanical processing(TMP).

Objectives of designing a deformation processing schedule for two-phase alloys therefore are :

(a) transformation of initial microstructure from an initial to a final microstructure

(b) assessing load requirements so that a given deformation processing schedule can be carried out with a given set of equipments

(c) assuring that the work piece material is workable under the given stress - strain - strain rate - temperature domain.

The present study is not concerned with the third objective.

1.2.1 Transformation of the initial microstructure to a final one

Mechanical working has been historically used as the primary means of changing the size and shape of materials while transforming the cast structure of an ingot into what is generally refered to as a wrought product. The two principal methods of working are forging and rolling along with other means such as extrusion and drawing. But at present it is also done for obtaining desired microstructure. Microstructural features of importance for property control include[3] :

(a) coherency and distribution of strengthening precipitates

(b) degree of recrystallization

(c) grain size and shape

(d) crystallographic texture

(e) size, shape and distribution of dispersed second phase (e.g. intermetallics).

Examples of incorporating thermo-mechanical working in the manufacturing schedule are controlled rolling of steels, thermo-mechanical processing of Al alloys and control of morphology of α-plates in $\alpha+\beta$ Ti alloys. During mechanical working of Ti-alloys in the $\alpha+\beta$ range, α-plates elongate with deformation. Finally they break into smaller plates which get recrystallized during subsequent heat-treatment into equiaxed structures. Fig.(1.3) shows the effect of strain due to forging and subsequent annealing on the length distribution of α-phase[7].

Roberts has reviewed the microstructural changes during hot deformation of coarse two-phase materials like $\alpha+\beta$ brass and $\alpha+\gamma$ stainless steel[8]. By coarse two-phase material we mean that both the phases have same grain-size. There have been very few systematic studies aimed at elucidating the microstructural evolution in association with hot deformation of two-phase alloys. In these studies it has been found that one microconstituent undergoes restoration through dynamic recovery plus recrystallization and the other softens via dynamic recovery only. It has been observed that orientation of plate like phase

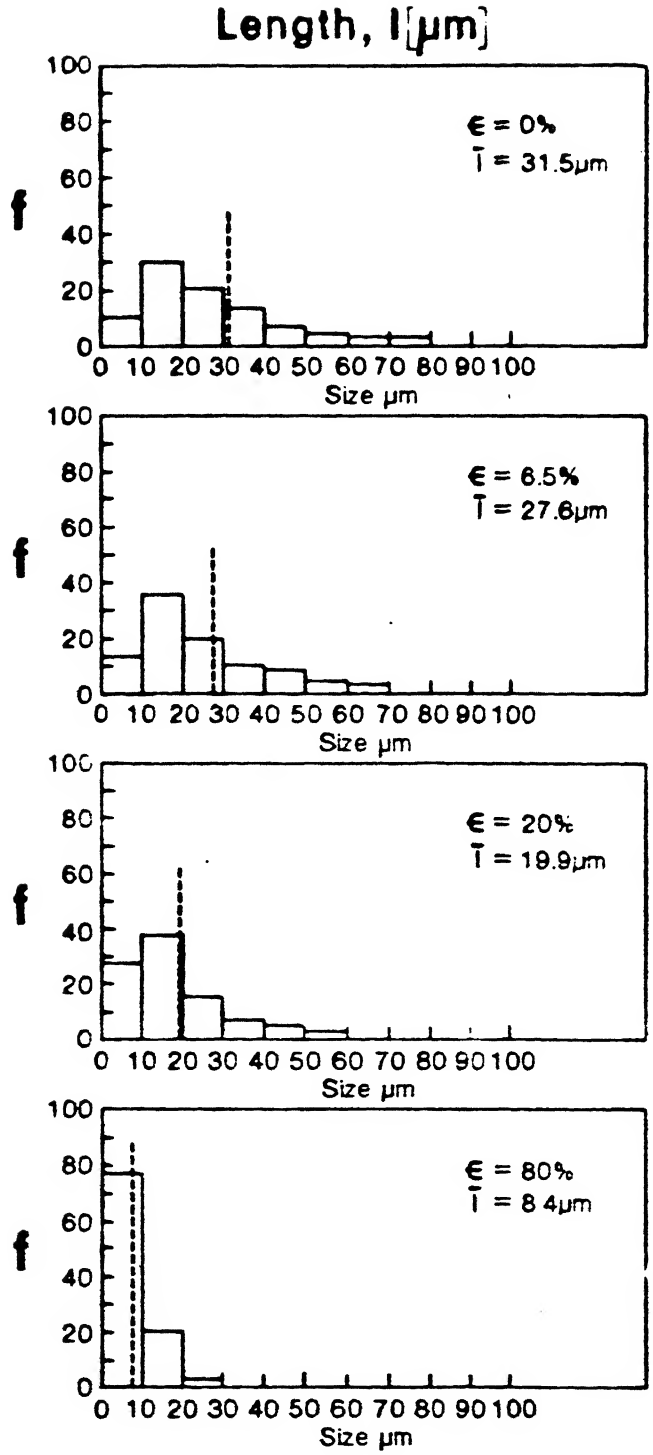


Fig.1.3 The effect of strain by forging and subsequent annealing on the length distribution of the α-phase, in material with initial thin lenticular structure

significantly changes with time during deformation processing.

1.2.2 Assessment of the load requirement

For designing a high temperature deformation processing schedule under given conditions it is very important to know the loads, size reduction and overall rate of deformation imparted to a two- phase alloy which will lead to the desired microstructure. Many theoretical studies using analytical[9] and computational[10] procedures have been made to meet these requirements in rolling, forging, extrusion and other metal working operations. These procedures, however, treat the material only at macro-level and microstructural variables have not been accounted for. However, a few efforts have been made in this direction in the recent past.

Gurland[1] has reviewed the application of the law of mixtures to the plastic deformation behaviour of two-phase alloys. He has shown that the law of mixture which states that,

$$\sigma_c = \sigma_\alpha V_\alpha + \sigma_\beta V_\beta \qquad \ldots(1.3a)$$

$$\varepsilon_c = \varepsilon_\alpha V_\alpha + \varepsilon_\beta V_\beta \qquad \ldots(1.3b)$$

where V is volume fraction, and σ and ε are the average values of the in-situ stresses and strains in the composite and in each of the two-phases respectively, is applicable for the plastic deformation of the two-phase alloys in the small strain range.

Sastry et al[6] have discovered that law of mixture is not applicable for the case of high temperature plastic deformation in $\alpha+\beta$ Ti alloys.

Ankem and Margolin[5] by using NASTRAN computer program and stress-strain curves for α and β phases of Ti-Mn alloys have calculated reasonably well the effect of particle size, matrix and volume fraction on stress-strain relations of the two-phase $\alpha+\beta$ Ti-alloys. They found that for a given volume fraction of α-phase the calculated stress-strain curve was higher for a finer particle size than for a coarse particle size and this behaviour was seen for all the alloys with different volume fraction of α-phase. The calculated stress-strain curves for four $\alpha+\beta$ alloys with different volume percent ofα-phase were compared with their corresponding experimental curves and in general good agreement was found. Discrepancies were attributed to lack of consideration of particle morphology.

Dragone and Nix have studied the creep behaviour of fiber reinforced metals using FEM[11]. They have modelled a fiber reinforced metals shown in Fig.(1.4a) whose creep deformation rate will depend on many factors like fibre volume fraction (V_f), fibre aspect ratio (L/D), degree of fibre overlap, fibre/matrix interface properties ($\sigma_{interface}$), relative fibre spacing (D/L) as simple unit cell shown in Fig.(1.4b). Their results are shown in Fig.(1.5). A study of this type can be useful for understanding the effect of morphology of individual phases on the overall high temperature plastic deformation of two-phase alloys. This is very important from processing point of view.

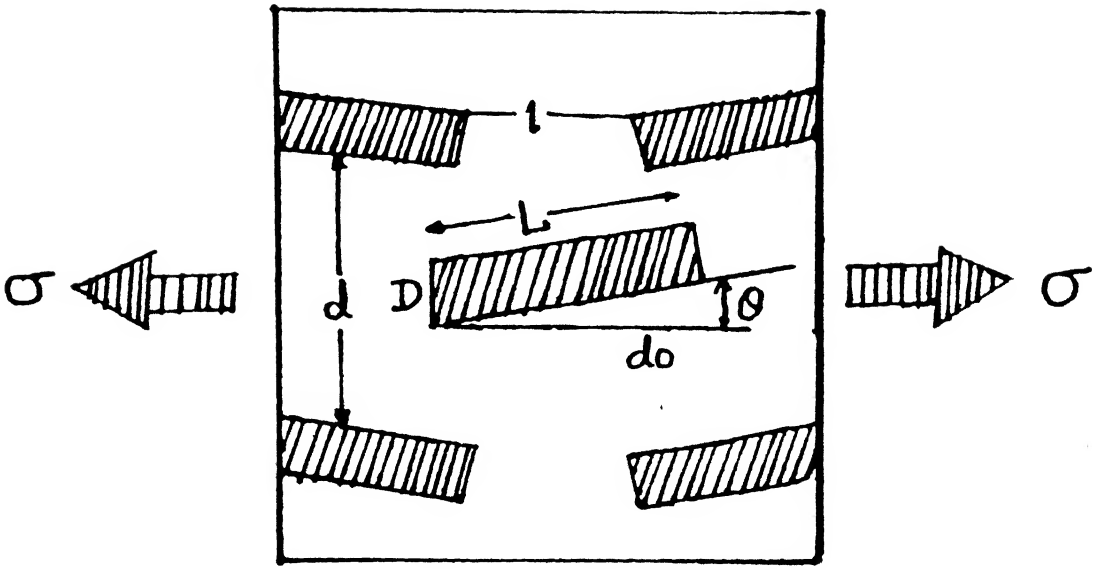


Fig.1.4a Schematic diagram of a reinforced metal showing important geometric parameters

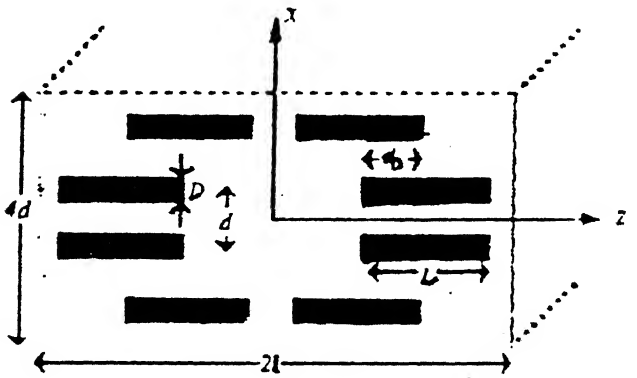


Fig.1.4b Development of the unit cell model

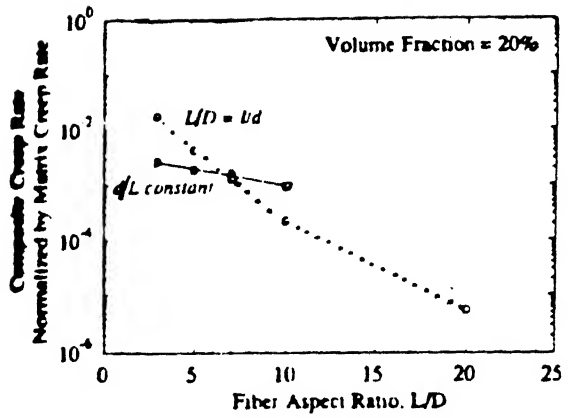


Fig.1.5a Variation in the calculated composite steady state creep rate with fiber aspect ratio at constant volume fraction for two different cases: (1) fiber aspect ratio varied while keeping unit cell aspect ratio (l/d) equal to fiber aspect ratio (L/D) and (2) fiber aspect ratio varied while keeping relative fiber spacing (d/L) constant

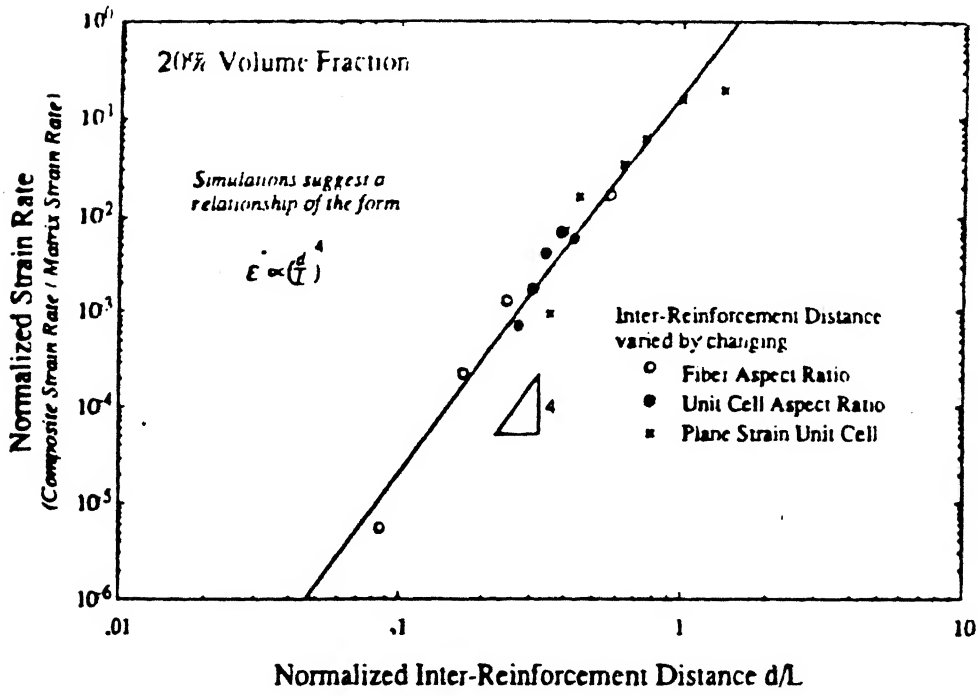


Fig.1.5b Variation in the calculated composite steady state creep rate with relative reinforcement spacing for various unit cell geometries. Both fiber and plate geometries are included on this plot

1.3 Limitations of Existing Analytical Procedures

A review of the literatures on deformation processing of two-phase material, therefore, indicates the following :

(a) most of the computational work associated with the load requirement during plastic deformation of two-phase structures has been carried out assuming material to behave like elasto-plastic. This work, though useful, is strictly applicable for low temperature deformation

(b) very little work has been done in connection with heterogeneities in stress, strain and strain-rate

(c) effect of shape of the second phase particles on the overall deformation behaviour has not been accounted for

(d) most investigators have analysed the overall deformation behaviour of material assuming that the second phase plates/fibers are aligned in a particular direction. This is applicable for aligned fiber composites but not for two-phase alloys

(e) in principle, overall deformation properties assigned should be invariant to $\dot{\varepsilon}$, ε fields. This check has not been carried out by most of the earlier investigators.

CHAPTER 2

FORMULATION OF THEORETICAL MODEL FOR THE HIGH TEMPERATURE PLASTIC DEFORMATION OF A TWO PHASE STRUCTURE

2.1 Introduction

As stated in the previous chapter, the response of a two-phase structure to the stress state imposed on it during its deformation processing depends on the plastic deformation behaviour of the aggregate which in turn is influenced by :

(i) physical and mechanical properties of constituent phases

(ii) their volume fractions, and

(iii) morphological features i.e. the shape, size and distribution of individual phases.

Due to the large number of variables which are usually needed for obtaining an explicit constitutive relationship for a given structure the plastic deformation behaviour of a two-phase aggregate is known to be extremely complex. Zaoui[12] has pointed out two possible approaches which in principle can be adopted to model the plastic behaviour of a given material. These are :

(a) the first one consists of phenomenological approach using internal variables whose physical meaning is not to be elucidated. These parameters are only expected to allow after adequate identification from standard experiments, a fairly good predicting ability i.e. fairly good

aggrement with further experimental results

(b) the second one proceeds from a deductive approach. It starts from some knowledge of physical phenomena involved in the macroscopic response and works out a mechanical description of related variables and laws of evolution. Then by dealing with available information (or adequate assumptions) on the initial microstructure of concerned material, it has to deduce the corresponding macroscopic behaviour through some homogenization procedure.

It is the second approach which has been adopted here. Clearly modelling of the deformation behaviour of a two-phase aggregate requires an understanding of deformation behaviour of individual constituents. The approach for the high temperature deformation of two-phase structure which has been adopted in the present analysis has been described in this chapter. This is preceeded by a general discussion on the fundamental considerations of the plasticity theory.

2.2 Plastic Deformation

2.2.1 Stress, strain and strain-rate tensors

The quantitave description of plastic deformation (as a matter of fact any type of deformation) is done by relating the stress tensor to strain and strain rate tensors. So, before reviewing the phenomenon of plastic deformation, stress, strain and strain-rate tensors will be defined.

State of stress in three-dimensions is described by a tensor of second rank, i.e. it consists of nine components, six of which are independent. Thus,

$$\sigma_{ij} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix} = \begin{bmatrix} \sigma_{XX} & \sigma_{XY} & \sigma_{XZ} \\ \sigma_{YX} & \sigma_{YY} & \sigma_{YZ} \\ \sigma_{ZX} & \sigma_{ZY} & \sigma_{ZZ} \end{bmatrix} \quad \ldots(2.1)$$

where $\sigma_{12} = \sigma_{21}$, $\sigma_{13} = \sigma_{31}$ and $\sigma_{23} = \sigma_{32}$

Like all other symmetric tensor quantities σ_{ij} can be split into two components namely (i) hydrostatic and (ii) deviatoric. Accordingly,

$$\sigma_{ij} = \sigma'_{ij} + \frac{1}{3} \sigma_{ij} \delta_{ij} \quad \ldots(2.2)$$

where σ'_{ij} is the deviatoric component and δ_{ij} is the Kronecker delta defined as,

$$\delta_{ij} = \begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix} = \begin{matrix} 1 & i = j \\ 0 & i \neq j \end{matrix} \quad \ldots(2.3)$$

It is clear from the above definition that the stress state at a point changes the orientation of coordinate axes. Invariant coefficients of stress tensor i.e. I_1, I_2, I_3 are obtained in the usual manner. They are as follows,

$$I_1 = \sigma_{XX} + \sigma_{YY} + \sigma_{ZZ} \qquad \ldots(2.4a)$$

$$I_2 = \sigma_{XX}\,\sigma_{YY} + \sigma_{YY}\,\sigma_{ZZ} + \sigma_{XX}\,\sigma_{ZZ} - \sigma^2_{XY} - \sigma^2_{XZ} - \sigma^2_{YZ} \qquad \ldots(2.4b)$$

and $$I_3 = \sigma_{XX}\,\sigma_{YY}\,\sigma_{ZZ} + 2\sigma_{XY}\,\sigma_{YZ}\,\sigma_{ZX} - \sigma_{XX}\,\sigma^2_{XZ} - \sigma_{ZZ}\,\sigma^2_{XY} \qquad \ldots(2.4c)$$

Similarly, strain tensor at a point is defined as

$$\varepsilon_{ij} = \begin{pmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & \varepsilon_{33} \end{pmatrix} = \begin{pmatrix} \varepsilon_{XX} & \varepsilon_{XY} & \varepsilon_{XZ} \\ \varepsilon_{YX} & \varepsilon_{YY} & \varepsilon_{YZ} \\ \varepsilon_{ZX} & \varepsilon_{ZY} & \varepsilon_{ZZ} \end{pmatrix} \qquad \ldots(2.5)$$

where $\varepsilon_{XX} = \dfrac{\partial \delta_X}{\partial X}$, $\varepsilon_{YY} = \dfrac{\partial \delta_Y}{\partial Y}$, $\varepsilon_{ZZ} = \dfrac{\partial \delta_Z}{\partial Z}$,

$$\varepsilon_{XY} = \varepsilon_{YX} = \frac{1}{2}\left(\frac{\partial \delta_X}{\partial Y} + \frac{\partial \delta_Y}{\partial Y}\right)$$

$$\varepsilon_{XZ} = \varepsilon_{ZX} = \frac{1}{2}\left(\frac{\partial \delta_X}{\partial Z} + \frac{\partial \delta_Z}{\partial X}\right)$$

$$\varepsilon_{YZ} = \varepsilon_{ZY} = \frac{1}{2}\left(\frac{\partial \delta_Z}{\partial Y} + \frac{\partial \delta_Y}{\partial Z}\right)$$

here δ_X , δ_Y and δ_Z are components of displacement vector along X, Y, and Z axes. Strain-rate is the rate of change of strain with time. Therefore, strain-rate tensor is defined as,

$$\dot{\varepsilon}_{ij} = \begin{pmatrix} \dot{\varepsilon}_{11} & \dot{\varepsilon}_{12} & \dot{\varepsilon}_{13} \\ \dot{\varepsilon}_{21} & \dot{\varepsilon}_{22} & \dot{\varepsilon}_{23} \\ \dot{\varepsilon}_{31} & \dot{\varepsilon}_{32} & \dot{\varepsilon}_{33} \end{pmatrix} = \begin{pmatrix} \dot{\varepsilon}_{XX} & \dot{\varepsilon}_{XY} & \dot{\varepsilon}_{XZ} \\ \dot{\varepsilon}_{YX} & \dot{\varepsilon}_{YY} & \dot{\varepsilon}_{YZ} \\ \dot{\varepsilon}_{ZX} & \dot{\varepsilon}_{ZY} & \dot{\varepsilon}_{ZZ} \end{pmatrix} \quad \ldots(2.6)$$

where $\dot{\varepsilon}_{XX} = \frac{\partial U}{\partial X}$, $\dot{\varepsilon}_{YY} = \frac{\partial V}{\partial Y}$, $\dot{\varepsilon}_{ZZ} = \frac{\partial W}{\partial Z}$

$$\dot{\varepsilon}_{XY} = \dot{\varepsilon}_{YX} = \frac{1}{2} \left(\frac{\partial U}{\partial Y} + \frac{\partial V}{\partial X} \right)$$

$$\dot{\varepsilon}_{XZ} = \dot{\varepsilon}_{ZX} = \frac{1}{2} \left(\frac{\partial U}{\partial Z} + \frac{\partial W}{\partial X} \right)$$

$$\dot{\varepsilon}_{YZ} = \dot{\varepsilon}_{ZY} = \frac{1}{2} \left(\frac{\partial W}{\partial Y} + \frac{\partial V}{\partial Z} \right)$$

here U, V and W are velocity components along X, Y and Z directions.

Strain and strain tensors also like stress tensor can be split into two components as hydrostatic and deviatoric.

$$\varepsilon_{ij} = \frac{1}{3} \varepsilon_{ii} \delta_{ij} + \varepsilon'_{ij} \quad \ldots(2.7)$$

where ε'_{ij} is deviatoric component of strain tensor and

$$\dot{\varepsilon}_{ij} = \frac{1}{3} \dot{\varepsilon}_{ii} \delta_{ij} + \dot{\varepsilon}'_{ij} \quad \ldots(2.8)$$

where $\dot{\varepsilon}'_{ij}$ is deviatoric component of strain-rate tensor.

Invariant coefficients of strain and strain-rate tensors are defined in the way similar to that followed for

stress-tensor. Effective stress ($\bar{\sigma}$), effective strain ($\bar{\varepsilon}$) and effective strain-rate ($\dot{\bar{\varepsilon}}$) are related to second invariant of their respective tensors and are expressed as,

$$\bar{\sigma} = \sqrt{(\sigma_{XX} - \sigma_{YY})^2 + (\sigma_{YY} - \sigma_{ZZ})^2 + (\sigma_{ZZ} - \sigma_{XX})^2 + 6(\sigma_{XY}^2 + \sigma_{YZ}^2 + \sigma_{XZ}^2)} \quad \ldots(2.9a)$$

$$\bar{\varepsilon} = \sqrt{(\varepsilon_{XX} - \varepsilon_{YY})^2 + (\varepsilon_{YY} - \varepsilon_{ZZ})^2 + (\varepsilon_{ZZ} - \varepsilon_{XX})^2 + 6(\varepsilon_{XY}^2 + \varepsilon_{YZ}^2 + \varepsilon_{XZ}^2)} \quad \ldots(2.9b)$$

and

$$\dot{\bar{\varepsilon}} = \sqrt{(\dot{\varepsilon}_{XX} - \dot{\varepsilon}_{YY})^2 + (\dot{\varepsilon}_{YY} - \dot{\varepsilon}_{ZZ})^2 + (\dot{\varepsilon}_{ZZ} - \dot{\varepsilon}_{XX})^2 + 6(\dot{\varepsilon}_{XY}^2 + \dot{\varepsilon}_{YZ}^2 + \dot{\varepsilon}_{XZ}^2)} \quad \ldots(2.9c)$$

2.2.2 Incompressibility

If it is assumed that during the course of plastic deformation of any material no cavitation or internal cracks are formed, the density of the material remains unaltered. Under such conditions the constancy of material volume is maintained. Therefore, the sum of all the diagonal components of strain tensor (ε_{ii}) , which represents volumetric change, is zero. Thus

$$\varepsilon_{ii} = \varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33} = 0 \quad \ldots(2.10)$$

This along with Eqs.2.7 and 2.8 lead to

$$\varepsilon_{ij} = \varepsilon'_{ij} \quad \ldots(2.11a)$$

and $\dot{\varepsilon}_{ij}=\dot{\varepsilon}'_{ij}$ ...(2.11b)

This has an important consequence in theories of plasticity which so far have been attempting to relate the deviatoric component of strain and strain-rate to deviatoric component of stress.

2.2.3 General constitutive equations

Plastic deformation in general can be expressed by constitutive equations of the type,

$$\bar{\sigma} = f(\bar{\varepsilon}, \dot{\bar{\varepsilon}}, M) \quad \ldots(2.12)$$

where M represents the microstructural state.

Most materials below the recrystallization temperature, i.e. in the cold working range, are not affected by moderate strain rates. Approximate stress-strain relationships for a limited region of strain can often be given by exponential equation of the form,

$$\bar{\sigma} = K(\bar{\varepsilon})^n \quad \ldots(2.13)$$

where K and n are constants.

Apart from the above relationship other constitutive relationships have also been suggested. These are summarized in Table 2.1.

It can be assumed that at high temperatures dynamic softening mechanisms occur concurrently and hence no work-hardening occurs in the material being deformed. The most commonly used expression for high temperature deformation which

is the case in the present study is,

$$\bar{\sigma} = K(\bar{\dot{\varepsilon}})^{m'} \qquad \ldots(2.14)$$

where m' is the strain-rate sensitivity. The above constitutive relationship suggests that the material behaves like a Newtonian fluid when $m' = 1$. For values of $m' \neq 1$, the material can be assumed to be undergoing a viscoplastic flow. Viscoplastic treatment of plastic deformation behaviour of a material is discussed in sections 2.3 and 2.4.2.

TABLE 2.1

Different Constitutive Relationships for Plastic Deformation in the Cold Working Range

Sl.No.	Author	Relationship
1	Unknown	$\bar{\sigma} = K(\bar{\varepsilon})^n$ K and n are constants
2	Ludwik	$\bar{\sigma} = a + b(\bar{\varepsilon})^c$ a, b, c are constants
3	Voce	$\bar{\sigma} = a + [b-a]\,[1-\exp(-C\bar{\varepsilon})]$ a, b, c are constants
4	Swift	$\bar{\sigma} = C[a+\bar{\varepsilon}]^n$

2.3 Viscoplasticity

It is well known that under tensile or compressive load a ductile material deforms elastically upto a certain strain

beyond which it starts yielding. This limiting strain is known as elastic limit and upto the elastic limit the stress and strain in most of the materials are related by Hooke's law. Beyond the elastic limit when the material yields plastically, the stress (i.e. flow stress) becomes a function of strain and strain rate. Some of the properties of the solid undergoing plastic deformation are similar to that of a Non-Newtonian fluid i.e. the relationship between the shear stress and shear strain rate is non-linear. In fact, it is possible to define a psuedo-viscosity for a plastically yielding material which is the ratio between the various shear stress and the shear strain rate components. In the limit of a perfectly elastic or rigid solid the viscosity assumes an infinitely large value.

2.4 Viscoplastic Deformation Model for a Two Phase Composite

2.4.1 Definition of the problem

As mentioned earlier the plastic behaviour of two phase aggregate is a complex one. One particular expression of, this complexity, strictly reflecting that of nature, consists in the large number of variables which are usually needed in order to yield an explicit constitutive law which could suffer any extensive comparison with experimental data for a given material[12]. Such a difficulty may be overcome by some homogenization procedure. By homogenization we mean finding out deformation properties of a homogeneous material whose deformation behaviour is same as the overall macroscopic deformation behaviour of the two phase aggregate. It can be done

in either of the two ways :

(a) microstructure is completely described and the solution may be an exact one

(b) microstructure is only partially and statistically defined.

First case is very difficult and impractical. A limited amount of work based on the second approach has been carried out so far[12] . In principle it is possible to work out many homogenization schemes.

In the microstructure of many alloys like Ti-alloys as well as in aligned fibre composites, plate like second phase structures are observed as shown in Fig.(2.1). The structure shown in Fig.(2.1) is very similar to that in Fig.(2.2). The deformation behaviour of the two phase aggregate shown in Fig.(2.2) can be very useful to pradict the overall macroscopic deformation behaviour of two phase aggregate which have plate like structure. As a first step, it was decided to study the deformation of one of the element of the aggregate. So, in the present investigation it was decided that a thorough study of a geometrically simple two phase composite, as shown in Fig.(2.3) can enlighten many features of two phase deformation. Investigation consists of modelling the deformation of a rectangular plate-like viscoplastic second phase particle embedded in a first phase viscoplastic matrix, within a square domain.

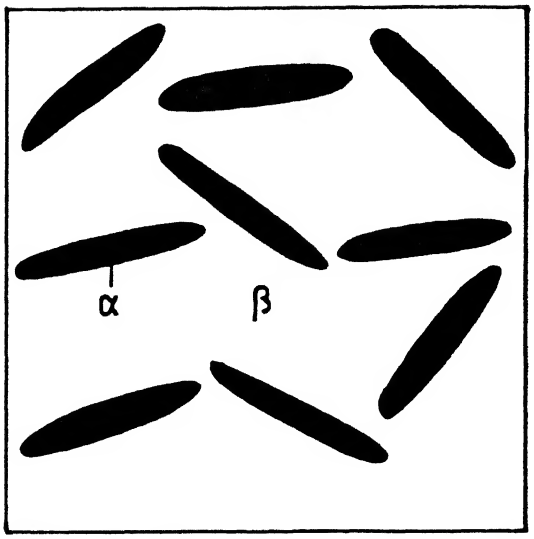


Fig.2.1 Schematic diagram of two phase (α+β) microstructure with plate-like α - phase.

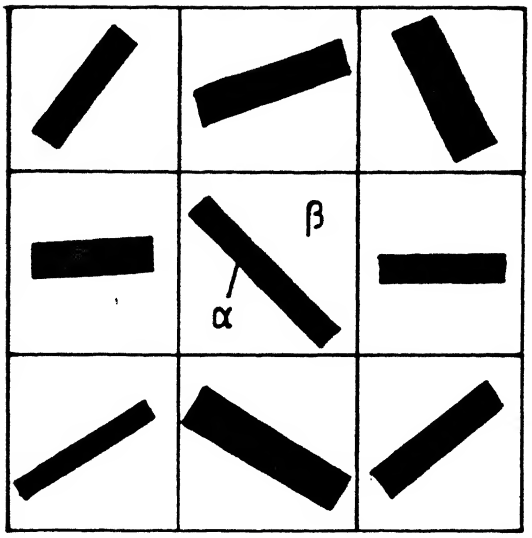


Fig.2.2 (α+β) microstructure (simplified)

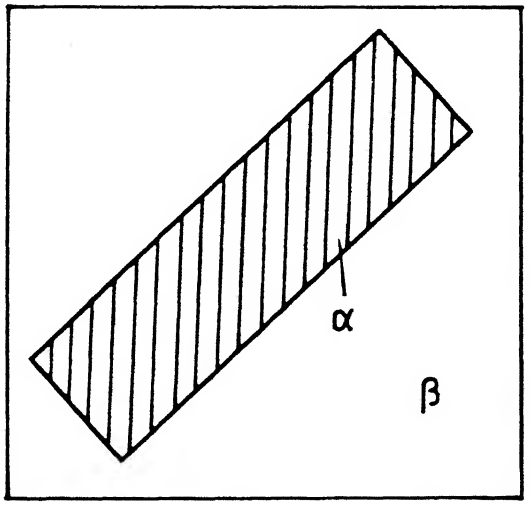


Fig.2.3 A unit cell taken from Fig.2.2

2.4.2 Constitutive relation for visco-plastic flow

Psuedo-Viscosity of metals at high temperature is dependent on local strain rates. The constitutive relation is

$$\sigma'_{ij} = 2\mu\, \dot{\varepsilon}'_{ij} \qquad \ldots(2.15)$$

where $\dot{\varepsilon}'_{ij}$ is the deviatoric part of strain-rate tensor. The strain-rate tensor is expressed in terms of velocity gradients in the material.

The viscosity μ for a strain-rate sensitive material is taken to be a function of uniaxial yield stress of the material (σ_Y) and the effective strain-rate ($\dot{\varepsilon}$).

$$\mu = f(\sigma_Y, \dot{\varepsilon})$$

$$= \frac{\sigma_Y + \left[\frac{\dot{\varepsilon}}{\sqrt{3}\gamma}\right]^{1/n}}{\sqrt{3}\,\dot{\varepsilon}} \qquad \ldots(2.16)$$

where γ and n are the physical constants which define the visco-plastic characteristics of the material. Values of γ and n usually lie between 1 and 2.

Flow stress or effective yield stress $\bar{\sigma}$ of the material is given by

$$\bar{\sigma} = \sigma_Y + \left[\frac{\dot{\varepsilon}}{\sqrt{3}\gamma}\right]^{1/n} \qquad \ldots(2.17)$$

By rearranging the terms in Eqs.2.16 and 2.17,

$$\bar{\sigma} = \sqrt{3}\,\mu\,\bar{\dot{\varepsilon}}$$

Or
$$\mu = \frac{\bar{\sigma}}{\sqrt{3}\bar{\dot{\varepsilon}}} \qquad \ldots(2.18)$$

But since $\bar{\sigma} = K\,\bar{\dot{\varepsilon}}^{m'}$

Therefore, $\mu = \frac{K}{\sqrt{3}}\,\bar{\dot{\varepsilon}}^{m'-1}$ $\qquad \ldots(2.19)$

2.4.3 Velocity fields in a deforming visco-plastic material

Constitutive relation for visco-plastic materials is given by Eqs.2.15 and 2.19.

From mass balance (or incompressibility condition)

$$\bar{\nabla} \cdot \vec{V} = 0$$

or

$$\frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y} = 0 \qquad \ldots(2.20)$$

for two-dimensional deformation.

From stress balance,

$$\rho(\vec{V} \cdot \nabla \cdot \vec{V}) = \nabla \cdot \underline{\sigma} + f \qquad \ldots(2.21)$$

where `f' is the body force and

$$\underline{\sigma} = \sigma_{XX}\,\hat{i}\hat{i} + \sigma_{XY}\,\hat{i}\hat{j} + \sigma_{YX}\,\hat{j}\hat{i} + \sigma_{YY}\,\hat{j}\hat{j} \qquad \ldots(2.22)$$

The stress balance equation can be thus written in the form

X direction : $$\rho\left(U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y}\right) = \left\{\frac{\partial}{\partial X}(\sigma_{xx}) + \frac{\partial}{\partial Y}(\sigma_{xy})\right\} \quad \ldots(2.23a)$$

Y direction : $$\rho\left(U\frac{\partial V}{\partial X} + V\frac{\partial V}{\partial Y}\right) = \left\{\frac{\partial}{\partial X}(\sigma_{xy}) + \frac{\partial}{\partial Y}(\sigma_{yy})\right\} \quad \ldots(2.23b)$$

The constitutive relations for 2-dimensional deformation can be written as follows.

$$\sigma_{xx} = -p + 2\mu\frac{\partial U}{\partial X} \quad \ldots(2.24a)$$

$$\sigma_{xy} = -\sigma_{yx} = \mu\left(\frac{\partial U}{\partial Y} + \frac{\partial V}{\partial X}\right) \quad \ldots(2.24b)$$

and $$\sigma_{yy} = -p + 2\mu\frac{\partial V}{\partial Y} \quad \ldots(2.24c)$$

where p is pressure or hydrostatic stress (or $\frac{\sigma_{ii}}{3}$)

Substituting the constitutive relations in Eqs.2.15 and 2.19 we get,

$$\rho\left(U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y}\right) = \left\{\frac{\partial}{\partial X}\left(-p + 2\mu\frac{\partial U}{\partial X}\right) + \frac{\partial}{\partial Y}\left[\mu\left(\frac{\partial U}{\partial Y} + \frac{\partial V}{\partial X}\right)\right]\right\} \ldots(2.25a)$$

$$\rho\left(U\frac{\partial V}{\partial X} + V\frac{\partial V}{\partial Y}\right) = \left\{\frac{\partial}{\partial Y}\left(-p + 2\mu\frac{\partial V}{\partial Y}\right) + \frac{\partial}{\partial X}\left[\mu\left(\frac{\partial U}{\partial Y} + \frac{\partial V}{\partial X}\right)\right]\right\} \ldots(2.25b)$$

The above two equations along with Eq.(2.20) form the set of governing equations describing the visco-plastic model.

2.4.4 Boundary conditions

Once the governing equations required to model the deformation of individual phases have been laid down, one needs the boundary conditions also. Boundary conditions that are required to solve the governing equations can be classified into three categories :

(a) essential boundary condition i.e., specifying U,P,V along the boundaries

(b) natural boundary condition i.e., specifying $\frac{\partial U}{\partial X}$, $\frac{\partial U}{\partial Y}$, $\frac{\partial V}{\partial X}$ and $\frac{\partial V}{\partial Y}$ along the boundaries

(c) traction boundary condition i.e., specifying t_x and t_y along the boundaries.

For the present investigation any of the above type of boundary condition can be used. For the sake of simplicity essential boundary conditions have been used.

2.4.5 Variable parameters

The next aspect of the model is the characterization of shape, size and relative orientation of plate-like second phase particles. Shape of the second phase plate-like particles can be conveniently described by L/D ratio where L is the length and D is the width. Further L x D gives the size of these particles (i.e. volume fraction).

While simulating this model these three parameters have been varied. The ratio $\frac{K_2}{K_1}$ where K_1 and K_2 stands for the values of constant K as defined in Eq.(2.14) for the matrix and the

plate-like second phase respectively, also have been varied quite extensively. Boundary condition is also one of the variable parameters that has been used in the present investigation. The value of m' has been kept constant for both the phases and set equal to 0.05. The range over which these parameters vary is given in Table 2.2.

TABLE 2.2

Range of Variation of Model Parameters

Sl. No.	Parameter	Range
1	volume fraction (V_f)	0.08 - 0.44
2	L/D	1.0 - 4.0
3	K_2/K_1	2.0 - 5.0
4	plate orientation (m)	0.0 - 7.0

2.5 Possible Approaches to Solve the Governing Equations

Analytical solutions for differential equations in engineering problems are available for very rare situations. Apart from that in the present investigation two phases with different material properties are there. It was decided to solve the governing equations with given boundary conditions

numerically which can be made possible with the help of computer. Out of various computational techniques for solving differential equations finite element method is one of the most powerful technique available and has been utilized in the present study. Finite element analysis of the present model is discussed in chapter 3.

CHAPTER 3

FINITE ELEMENT ANALYSIS

3.1 Introduction

As stated in chapter 2 to model the deformation of a two-phase alloy at high temperature, the differential Eqs.2.20, 2.25a and 2.25b, which governs the flow of viscoplastic material have to be solved. Finite element technique employing Galerkin's weighted residual method used to meet this objective. In this chapter first one of the numerical techniques used in our investigation, weighted residual method has been discussed. Then finite element procedure has been outlined and its application for solving the governing differential equations has been outlined. At the end of this chapter the essential features of the programming have been presented.

3.2 Weighted Residual Method

A differential equation should be satisfied at every point in the solution domain. However the numerical solution for the problems may not satisfy the governing equations exactly. It may satisfy the equations approximately having a small non-zero residue at most of the regions in the solution domain.

Let ϕ be the exact solution and ϕ^* be an approximate solution. Substituting these solutions in the differential equation yields

$$\left.\begin{array}{ll} L(\phi) = 0 & \text{Exact solution} \\ L(\phi^*) = R & \text{Approximate solution} \end{array}\right\} \quad \ldots(3.1)$$

where L is the differential operator of the problem. R is essentially a function whose value is equal or close to zero throughout the domain. Therefore this residue function is integrated over the domain. To pin-down the residue to small value at chosen number of points in the domain approximate weighting functions are multiplied during the minimization of residue in the whole solution domain.

The above discussed objectives are acheived by setting,

$$\int_D w_i R \, dV = 0 \qquad \ldots(3.2)$$

for i=1,2,.....n

where w_i are chosen weighting functions and D is the solution domain.

Substituting equation 3.1 in 3.2 it gives

$$\int_D w_i L(\phi^*) \, dV = 0 \qquad \ldots(3.3)$$

for i=1,2,.....n

Equation 3.3 form the basis of all weighted residual methods for solving differential equations. Higher the value of n more accurate is the solution. The manner in which the weighting functions are selected and the approximate solution ϕ^* is defined leads to many different weighted residual methods such as Ritz method, Galerkin method, Collocation method and the method of least squares. The Galerkin's method has a general applicability and hence been adopted here for the finite element solution procedure.

For Galerkin's method

$$W_i = N_i \qquad \ldots(3.4)$$

$$i=1,2,\ldots\ldots n$$

where N_i are shape functions of an element and n is the number of node points in the element.

3.3 Finite Element Procedure

The finite element as the name indicates, implies that the solution domain is divided into many finite sub-domains i.e. elements. This discretization is advantageous particularly for complex domains as by considering the element as a building block complicated domain shapes can be represented. This is possible even if the elements are of any particular shape.

The steps involved in any FEM solution are as follows :

(1) discretization of domain into small elements

(2) derivation of elemental properties

(3) grouping the elements into a global assembly

(4) incorporation of boundary conditions

(5) solving the resulting global matrix equations to obtain the field variables

(6) any processing that can further be done.

These steps are discussed in detail in the later part of this chapter. The advantage of splitting the domain into many elements is that the field variables can be interpolated using standard interpolation functions within each element. For this purpose nodes are selected within each element and the unknown variable is expressed in terms of the values of field variable at

the selected nodes. The interpolating functions for an element are also called shape functions. Every node has a shape function whose value is unity at that particular node and zero at all the other nodes of that element. This is a common property of all the shape functions although in their form they may differ from each other.

Mathematically the value of a field variable within an element can be expressed as,

$$\phi^* = \sum_{i=1}^{n} N_i \phi_i \qquad \ldots(3.5)$$

where ϕ_i are nodal values of the field variable.

Since such representations are possible for all the elements in the domain it can be said that the field variable is known in the entire domain.

3.4 Application of FEM in the Two Phase Model

3.4.1 Domain discretization

Discretization of domain involves division of the solution domain into small elements. For plane two-dimensional problems, these are n-sided polygons and hence many types of elements are possible like triangular, quadrilateral etc. The domain discretization seemingly simple, can affect the accuracy of the solution considerably, if done improperly. This is more important when the domain is of complex shape. Isoparametric elements (so-called because the field variable and the spatial coordinates can be defined with the help

of the same set of interpolation functions) offer many advantages and thus used in the present investigation. Eight noded quadrilateral elements as shown in Fig.(3.1) have been used. The domain with elements is shown in Fig.(3.2).

3.4.2 Derivation of finite element equations

Governing equations for the methods are given in chapter 2. Employing the Galerkin weighted residual method we get the following equations

$$\sum_{1}^{n^e} \int_{D^e} N_i \left\{ \rho\left(U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y}\right) + \frac{\partial P}{\partial X} - \mu\left(\frac{\partial^2 U}{\partial X^2} + \frac{\partial^2 U}{\partial Y^2}\right) \right\} dA = 0 \qquad \ldots(3.6)$$

$$\sum_{1}^{n^e} \int_{D^e} N_i \left\{ \rho\left(U\frac{\partial V}{\partial X} + V\frac{\partial V}{\partial Y}\right) + \frac{\partial P}{\partial Y} - \mu\left(\frac{\partial^2 V}{\partial X^2} + \frac{\partial^2 V}{\partial Y^2}\right) \right\} dA = 0 \qquad \ldots(3.7)$$

$$\sum_{1}^{n^e} \int_{D^e} M_l \left(\frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y}\right) dA = 0 \qquad \ldots(3.8)$$

where D^e represents the domain of an element, ne is the number of elements in the domain, i varies from 1 to 8 and l varies from 1 to 4. The velocity variables U and V and pressure P are interpolated within each element through the expressions,

$$U = \sum_{i=1}^{8} N_i U_i \;, \quad V = \sum_{i=1}^{8} N_i V_i \quad \text{and} \quad P = \sum_{l=1}^{4} M_l P_l \qquad \ldots(3.9)$$

where N_i are quadratic interpolation functions and M_l are linear interpolation functions in 2-D space. Lower order interpolation

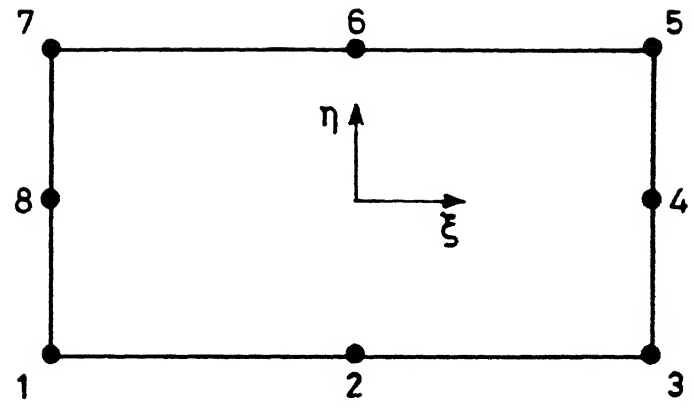


Fig. 3.1(a) Eight noded element showing the local co-ordinate system.

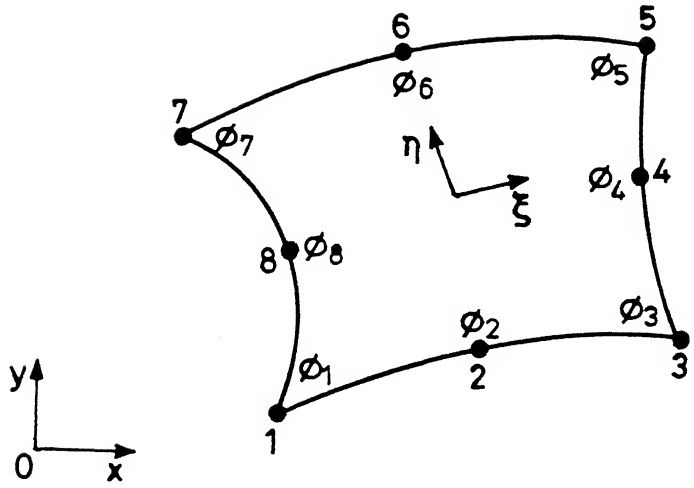


Fig. 3.1(b) Parabolic eight noded element in the physical domain.

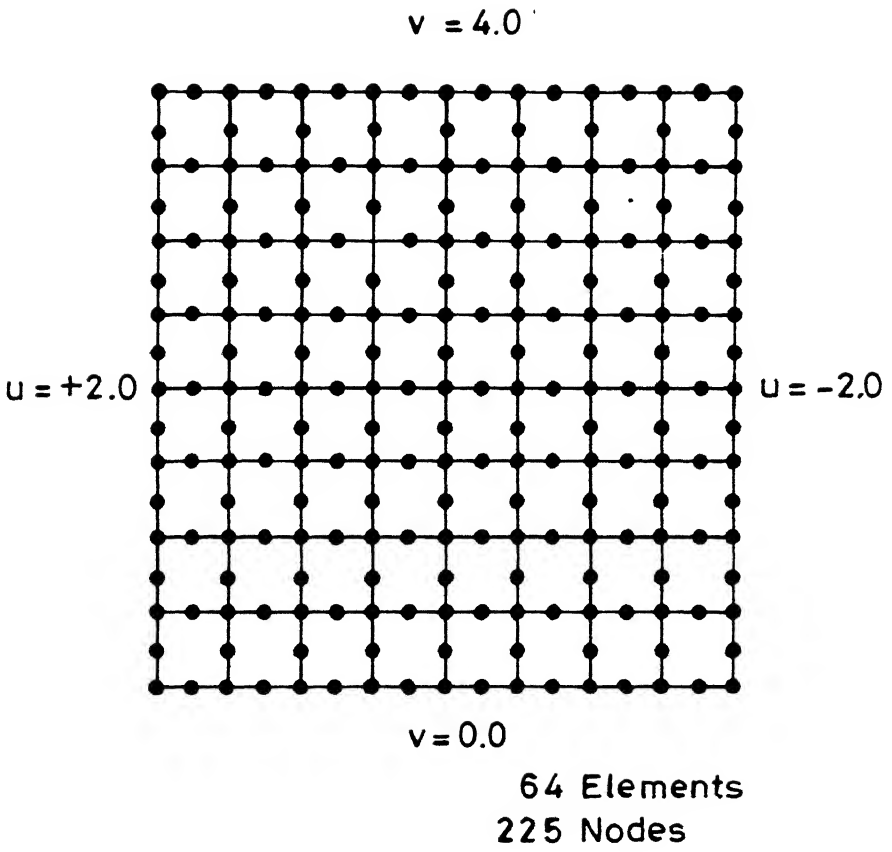


Fig. 3.2 Finite element mesh for the solution domain.

for pressure has been done for the sake of numerical stability. The expressions for N_i and M_i are presented subsequently. The equations 3.6 and 3.7 are respectively usedfor the solution of U-velocity and V-velocity components. For the solutions of pressure variables, the incompressibility equation 3.8 is used in an indirect manner by minimizing its residue with respect to the pressure interpolation functions M_i. Taken together, equations 3.6 to 3.8 are employed to obtain the nodal velocity and pressure variables.

Once the concept of subdividing a physical domain into elements is accepted the problem of depicting a variation in the variables(U,V,P) across the whole domain becomes far simpler since the variation can now be related within each element. Given such a variation within an element and the fact that elements are interconnected, it is a simple matter to visualise that the variation across the whole domain occurs in a piecewise manner. Interpolation functions which are generally polynomial are used to describe a variable within an element. The type of elements used to discretize the domain depends on the nature of the interpolation functions. For simplicity of integration and programming field variables as well as spatial coordinates are both described using interpolation functions of local coordinates. When same type of polynomial functions are used for describing both element geometry and variation of field variables, the element is called an isoparametric element.

3.4.3 Isoparametric transformation

Any field variable can be interpolated.

Let

$$\phi = \alpha_1 + \alpha_2\xi + \alpha_3\eta + \alpha_4\xi\eta \qquad \ldots(3.10)$$

be variation of ϕ with ξ,η (local coordinates) in a four noded rectangular element. A natural extension of Eq.(3.10) is

$$\begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{Bmatrix} = \begin{bmatrix} 1 & \xi_1 & \eta_1 & \xi_1\eta_1 \\ 1 & \xi_2 & \eta_2 & \xi_2\eta_2 \\ 1 & \xi_3 & \eta_3 & \xi_3\eta_3 \\ 1 & \xi_4 & \eta_4 & \xi_4\eta_4 \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{Bmatrix} \qquad \ldots(3.11)$$

where ϕ_1, ϕ_2, ϕ_3 and ϕ_4 are values of field variable ϕ and (ξ_1,η_1), (ξ_2,η_2), (ξ_3,η_3) and (ξ_4,η_4) are values of (ξ,η) at the four nodes of the element.

Eq.(3.11) can be rewritten as

$$\{\phi\}^e = \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{Bmatrix} = [C]\{\alpha\} \qquad \ldots(3.12)$$

Therefore $\{\alpha\} = [C]^{-1}\{\phi\}^e$ $\qquad \ldots(3.13)$

Also $$\phi = [1 \;\; \xi \;\; \eta \;\; \xi\eta] \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{Bmatrix} \qquad \ldots(3.14)$$

Therefore $\phi = [1\ \xi\ \eta\ \xi\eta]\ [C]^{-1}\ \{\phi\}^{e}$ ...(3.15)

or $$\phi = [N_1'\ N_2'\ N_3'\ N_4'] \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{Bmatrix} \quad ...(3.16)$$

where N_1', N_2', N_3' and N_4' are called shape functions of this particular element. Each of the functions correspond to the number of node given in their subscript. They are equal to unity at the corresponding nodes and equal to zero at other nodes.

For the present investigation isoparametric parabolic elements as shown in Fig.(3.1) has been used.

For ease of computation, as stated earlier, the shape functions have been defined in terms of local coordinates. Two types of shape function N_i (quadratic) and M_i (lower order interpolation) used for describing variation of variables U,V and P respectively are

(a) Corner nodes:

$$N_i = \frac{1}{4}(1 + \xi_i\xi)(\xi_i\xi + \eta_i\eta - 1)(1 + \eta_i\eta) \quad ...(3.17)$$

Mid side nodes :

$$\left.\begin{aligned} N_i &= \frac{1}{2}(1 - \xi^2)(1 + \eta_i\eta),\ \xi_i = 0 \\ N_i &= \frac{1}{2}(1 + \xi_i\xi)(1 - \eta^2),\ \eta_i = 0 \end{aligned}\right\} \quad ...(3.18)$$

where 'i' is the number of nodes and (ξ_i, η_i) are coordinates of the respective nodes in the local coordinate system.

(b) Corner nodes:

$$M_l = \frac{1}{4} (1 + \xi_l \xi) (1 + \eta_l \eta) \qquad \ldots(3.19)$$

where l is the number of corner nodes which varies from 1 to 4.

It has been stated earlier that in the present investigation isoparametric elements have been used. Accordingly,

$$\left.\begin{aligned} X &= \sum_{i=1}^{8} N_i X_i \\ Y &= \sum_{i=1}^{8} N_i Y_i \end{aligned}\right\} \qquad \ldots(3.20)$$

where (X_i, Y_i) are spatial coordinates of the nodes.

Comming back to Galerkin weighted residual method, using Eq.3.19 the X-momentum equation becomes

$$\sum_{1}^{ne} \int_{D^*} N_i \left[\rho \sum_{K=1}^{n} N_K U_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial X} U_j + \rho \sum_{K=1}^{n} N_K V_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial Y} U_j \right.$$

$$\left. + \sum_{l=1}^{m} \frac{\partial M_l}{\partial X} P_l - \mu \left[\sum_{j=1}^{n} \frac{\partial^2 N_j}{\partial X^2} U_j + \sum_{j=1}^{n} \frac{\partial^2 N_j}{\partial Y^2} U_j \right] \right] dA^* = 0$$

$$\ldots(3.21)$$

and Y- momentum equation becomes

$$\sum_{i}^{ne} \int_{D^e} N_i \left[\rho \sum_{K=1}^{n} N_K U_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial X} V_j + \rho \sum_{K=1}^{n} N_K V_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial Y} V_j \right.$$

$$\left. + \sum_{l=1}^{m} \frac{\partial M_l}{\partial Y} P_l - \mu \left[\sum_{j=1}^{n} \frac{\partial^2 N_j}{\partial X^2} V_j + \sum_{j=1}^{n} \frac{\partial^2 N_j}{\partial Y^2} V_j \right] \right] dA^e = 0$$

...(3.22)

Invoking Green's theorem the second order terms can again be reducd to the following expression :

$$\int_{D^e} N_i \left[\sum_{j=1}^{n} \frac{\partial^2 N_j}{\partial X^2} U_j + \sum_{j=1}^{n} \frac{\partial^2 N_j}{\partial Y^2} U_j \right] dA^e =$$

$$\int_{\Gamma^e} N_i \sum_{j=1}^{n} \frac{\partial N_j}{\partial \eta} U_j d\Gamma - \int_{A^e} \left[\frac{\partial N_i}{\partial X} \sum_{j=1}^{n} \frac{\partial N_j}{\partial X} U_j + \frac{\partial N_i}{\partial Y} \sum_{j=1}^{n} \frac{\partial N_j}{\partial Y} U_j \right] dA^e$$

...(3.23)

Also the terms $\left(\sum_{K=1}^{n} N_K U_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial X} U_j \right)$ and $\left(\sum_{K=1}^{n} N_K V_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial X} V_j \right)$ are

making the equations non-linear. So a guess value has been assumed for U_K and V_K and compared with calculated values of U and V. This leads to an iterative procedure. Therefore X-momentum equation becomes :

$$\sum_{1}^{ne} \int_{D^e} \left[N_i \rho \sum_{K=1}^{n} N_K \tilde{U}_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial X} U_j + N_i \rho \sum_{K=1}^{n} N_K \tilde{V}_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial Y} U_j \right.$$

$$\left. + N_i \sum_{l=1}^{m} \frac{\partial M_l}{\partial X} P_l + \mu \left[\frac{\partial N_i}{\partial X} \frac{\partial N_j}{\partial X} U_j + \frac{\partial N_i}{\partial Y} \frac{\partial N_j}{\partial Y} U_j \right] \right] dA^e$$

$$- \int_{\Gamma_1^e} \mu N_i \frac{\partial N_j}{\partial n} U_j \, d\Gamma - \int_{\Gamma_2^e} \mu N_i \left(\frac{\partial U_j}{\partial n} \right) d\Gamma = 0 \qquad \ldots(3.24)$$

and Y-momentum equation becomes

$$\sum_{1}^{ne} \int_{D^e} \left[N_i \rho \sum_{K=1}^{n} N_K \tilde{U}_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial X} V_j + N_i \rho \sum_{K=1}^{n} N_K \tilde{V}_K \sum_{j=1}^{n} \frac{\partial N_j}{\partial Y} V_j \right.$$

$$\left. + N_i \sum_{l=1}^{m} \frac{\partial M_l}{\partial Y} P_l + \mu \left[\frac{\partial N_i}{\partial X} \frac{\partial N_j}{\partial X} V_j + \frac{\partial N_i}{\partial Y} \frac{\partial N_j}{\partial Y} V_j \right] \right] dA^e$$

$$- \int_{\Gamma_1^e} \mu N_i \frac{\partial N_j}{\partial n} V_j \, d\Gamma - \int_{\Gamma_2^e} \mu N_i \left(\frac{\partial V_j}{\partial n} \right) d\Gamma = 0 \qquad \ldots(3.25)$$

where $\tilde{U}_K$ and $\tilde{V}_K$ are guess values (usually taken as previous

iteration value or initial guess). For the continuity equation M_i is used as weighting function. So

$$\sum_{1}^{ne} \int_{D^e} M_i \left(\frac{\partial N_j}{\partial X} U_j + \frac{\partial N_j}{\partial Y} V_j \right) dA^e = 0 \qquad \ldots(3.26)$$

The interesting point to note in equations 3.24, 3.25 and 3.26 is that there are twenty variables namely U and V at eight nodes and pressure at four corner nodes and there are twenty simultaneous equations per element.

Now, the next step is to calculate the coefficients associated with the twenty unknowns in these equations. For calculating the coefficients integration of certain functions have to be carried out. So the concepts of numerical integration have been reviewed.

3.4.4 Numerical integration

A numerical integration procedure is adopted where the sampling points are used and termed as Gauss-points. In particular three point Gauss-Legendre quadrature has been used which leads to high accuracy. The unidimensional example as shown is Fig.(3.3) illustrates discrete sampling points and, generally, the Gaussian quadrature rule leads to an equation of the form

$$I = \int_{-1}^{+1} f(\xi) d\xi = \sum_{i=1}^{m} a_i f(\xi_i) \qquad \ldots(3.27)$$

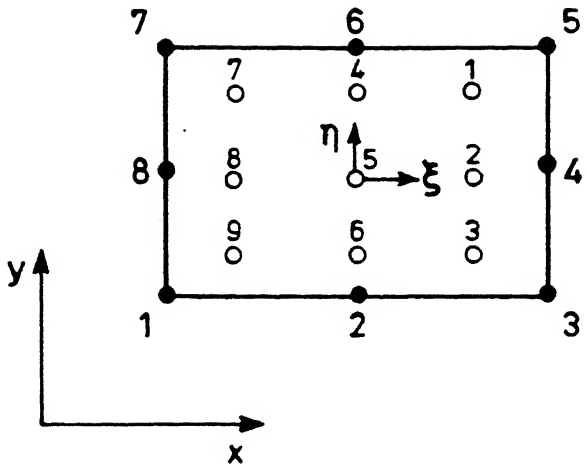


Fig.3.3 Nodal definition of a parabolic element with a 3x3 Gauss point scheme.

where m = total number of integration points

a_i = ith weighting factor

ξ_i = coordinate of the ith integration point

For two dimensional case

$$I = \int_{-1}^{+1}\int_{-1}^{+1} F(\xi,\eta)\,d\xi\,d\eta$$

$$= \sum_{j=1}^{m}\sum_{i=1}^{m} a_j a_i\, F(\xi_i,\eta_i) \qquad \ldots(3.28)$$

Clearly from equations 3.24, 3.25 and 3.26 if a local coordinate system is to be used then the first order variations with respect to the spatial coordinates must also be expressed in local terms. This can be acheived by chain rule.

$$\left.\begin{aligned}\frac{\partial N_i}{\partial \xi} &= \frac{\partial N_i}{\partial X}\frac{\partial X}{\partial \xi} + \frac{\partial N_i}{\partial Y}\frac{\partial Y}{\partial \xi}\\ \frac{\partial N_i}{\partial \eta} &= \frac{\partial N_i}{\partial X}\frac{\partial X}{\partial \eta} + \frac{\partial N_i}{\partial Y}\frac{\partial Y}{\partial \eta}\end{aligned}\right\} \qquad \ldots(3.29)$$

Which may be rewritten in a more convenient from.

$$\begin{Bmatrix}\frac{\partial N_i}{\partial \xi}\\ \frac{\partial N_i}{\partial \eta}\end{Bmatrix} = \begin{bmatrix}\frac{\partial X}{\partial \xi} & \frac{\partial Y}{\partial \xi}\\ \frac{\partial X}{\partial \eta} & \frac{\partial Y}{\partial \eta}\end{bmatrix}\begin{Bmatrix}\frac{\partial N_i}{\partial X}\\ \frac{\partial N_i}{\partial Y}\end{Bmatrix} = [J]\begin{Bmatrix}\frac{\partial N_i}{\partial X}\\ \frac{\partial N_i}{\partial Y}\end{Bmatrix} \qquad \ldots(3.30)$$

Or

$$\begin{Bmatrix} \frac{\partial N_i}{\partial \xi} \\ \\ \frac{\partial N_i}{\partial \eta} \end{Bmatrix} = [J]^{-1} \begin{Bmatrix} \frac{\partial N_i}{\partial X} \\ \\ \frac{\partial N_i}{\partial Y} \end{Bmatrix}$$

From equations 3.20 and 3.30

$$[J] = \begin{bmatrix} \sum_{i=1}^{8} \frac{\partial N_i}{\partial \xi} X_i & \sum_{i=1}^{8} \frac{\partial N_i}{\partial \xi} Y_i \\ \\ \sum_{i=1}^{8} \frac{\partial N_i}{\partial \eta} X_i & \sum_{i=1}^{8} \frac{\partial N_i}{\partial \eta} Y_i \end{bmatrix}$$

Therefore $$[J]^{-1} = \frac{1}{\det[J]} \begin{bmatrix} \frac{\partial Y}{\partial \eta} & -\frac{\partial Y}{\partial \xi} \\ \\ -\frac{\partial X}{\partial \eta} & \frac{\partial X}{\partial \xi} \end{bmatrix}$$

Where $$\det[J] = \frac{\partial X}{\partial \xi}\frac{\partial Y}{\partial \eta} - \frac{\partial X}{\partial \eta}\frac{\partial Y}{\partial \xi}$$

Another transformation that is required integration is

$$dX\ dY = \det[J]\ d\xi\ d\eta$$

3.4.5 Element assembly

In section 3.4.3 the elemental contributions to the left hand side coefficient matrix and right hand side vector were discussed. The elemental contributions are assembled into global matrix by adding the entries corresponding to each nodal variable. We get

$$[A]\,[X] = [B] \qquad \ldots(3.36)$$

where [A] = Coefficient matrix

[X] = Solutiuon vector

[B] = Right hand side vector

Eq.(3.36) has been solved by Frontal method which has been described in the next section.

There are two important features related to calculation of coefficients that should be noted

(1) $\sum N_K \tilde{U}_K$ and $\sum N_K \tilde{V}_K$ are actually obtained as guess in the first iteration

(2) [A] is calculated using 3-point gauss rule. And μ for each gauss point within an element is different.

$$\mu = K\,\bar{\dot{\varepsilon}}^{m-1}$$

The value of K (which depends on whether it lies on the matrix or on the second phase plate) and $\bar{\dot{\varepsilon}}$ (as calculated in previous iteration) is different at all gauss points.

3.5 Programming

3.5.1 Salient features of the program

The computer program is based on the

algorithm developed by Taylor and Hughes[13]. A very simple flow chart of the algorithm is depicted in Fig.(3.4).

The main program calls the subroutines DIMENS, DINPUT, DRIVES, ITERAT, MATPRT and SRCAL which calls other subroutines.

The functions of the various subroutines are as follows:

(a) DIMENS returns the value of the array dimensions which remain unchanged through out the program.

(b) Subroutine DINPUT reads and returns all of the required input parameter which are listed below :

(1) slope of the second phase plate(m), L, D and $\frac{K_2}{K_1}$

(2) axi-symmetric flow indicator, the number of initial and boundary conditions, body forces, tolerance, relaxation parameters K_{matrix} (or K_1), density

(3) boundary conditions.

The subroutine DINPUT also does mesh generation by calling the subroutine MESANG. Also some of the given input is checked for correction by two routines named DIAGN1 and DIAGN2.

(c) Then subroutine MATPRT is called. Using the values of m, L and D it assigns the value of variable valt at different gauss-points if the gauss-point lies within the domain of second phase plate it is assigned $\frac{K_2}{K_1}$. Otherwise it is assigned the value one.

(d) After this subroutine DRIVES is called. It calculates N_i, M_l, $\frac{\partial N_i}{\partial X}$, $\frac{\partial N_i}{\partial Y}$, $\frac{\partial M_l}{\partial X}$, $\frac{\partial M_l}{\partial Y}$ and det[J] at all the gauss-points in all the elements. It further calls the

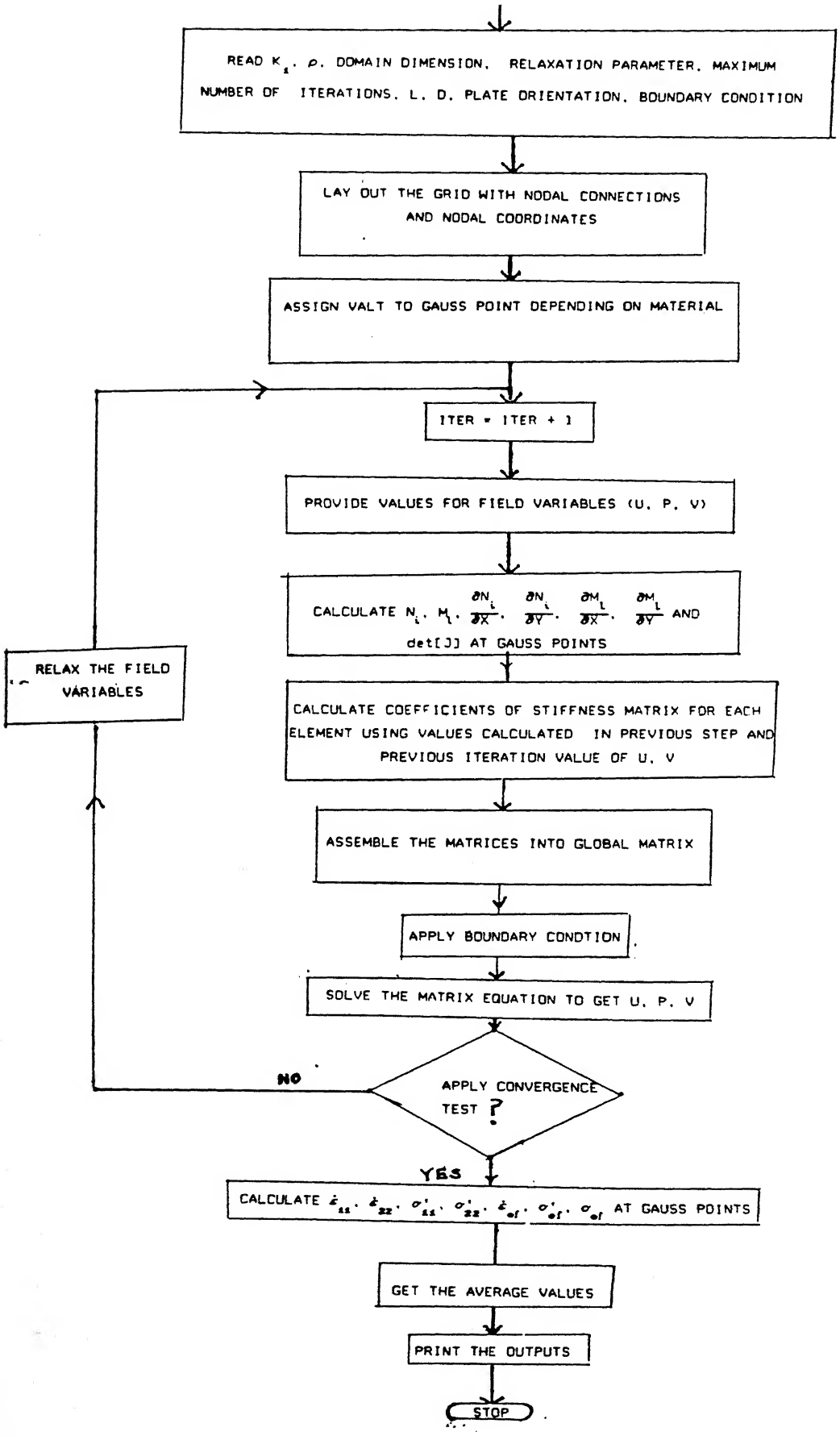


Fig.3.4 Flow chart for finite element procedure

subroutines SHAPE4, SHAPE8 and DJACON. SHAPE4 and SHAPE8 calculated M_l and N_i at the gauss-points. DJACOB calculates the values of det[J] $\frac{\partial N_i}{\partial X}$, $\frac{\partial N_i}{\partial Y}$, $\frac{\partial M_l}{\partial X}$ and $\frac{\partial M_l}{\partial Y}$ at gauss-points.

(e) After this the subroutine ITERAT is called which calls subroutines PRESCR, FORNTS, WRITER and TOLREL. Subroutine PRESCR is called when gradient of variables such as $\frac{\partial U}{\partial X}$, $\frac{\partial V}{\partial X}$ are specified in the boundary condition. It calls another subroutine SURFIN. Since gradient boundary condition is not used in the present investigation the subroutine PRESCR has no use.

Next subroutine that ITERAT calls is FRONTS. FRONTS calls subroutine MATRIX, which calculates (20 x 20) coefficient matrix for a given element. It uses the values of N_i, M_l, $\frac{\partial N_i}{\partial X}$, $\frac{\partial N_i}{\partial Y}$, $\frac{\partial M_l}{\partial X}$, $\frac{\partial M_l}{\partial Y}$, det[J], valt and previous guess values of field variables (U,P,V and $\dot{\varepsilon}$ which is calculated using U and V) to calculate coefficients.

In FRONTS coefficient matrices for all elements are assembled, boundary conditions applied and matrix equation is finally solved using frontal method to get the value of field variable (U,P,V). This is discussed in details in section .

These values are then printed using subroutine WRITER. TOLREL compares the values of calculated field variables with the previous iteration values and checks whether convergence criteria is satisfied. If covergence criteria is not satisfied then it relaxes the values for the next iteration.

(f) Finally subroutine SRCAL calculates the values of

$\dot{\epsilon}_{11}$, $\dot{\epsilon}_{22}$, $\dot{\epsilon}_{12}$, $\dot{\epsilon}_{ef}$, σ'_{ef}, σ'_{11}, σ'_{22} σ'_{12}, σ_{eff} at all gauss point and finds out their averages.

The general program structure is shown in Fig.(3.5).

3.5.2 Matrix solution procedure

Here the frontal solution method has been used to solve the assembled matrix equations. It is a wll known fact that the method adopted for solving the assembled matrix equation has a significant bearing on the computer storage requirement and execution time when the total number of unknown variables is very large. It becomes very time consuming and the core memory problem is very acute when a large number of unknowns are solved by the matrix inversion technique. The frontal solution technique can handle quite a large number of variables without the problem of large core memory and the computer execution time is also quite less.

The frontal solution technique used in the present work is based on the direct Gaussian elimination procedure for solving the symmetric matrices where the leading diagonal is always used as pivot. For unsymmetric matrices encountered in a wide range of engineering problems, the most suitable pivot is not necessarily on the leading diagonal and a Frontal solution routine exists for off diagonal pivoting. But since this tends to be more time consuming, the method used in the present work uses only diagonal pivoting and incorporates many features of the Frontal method for solving symmetric matrices.

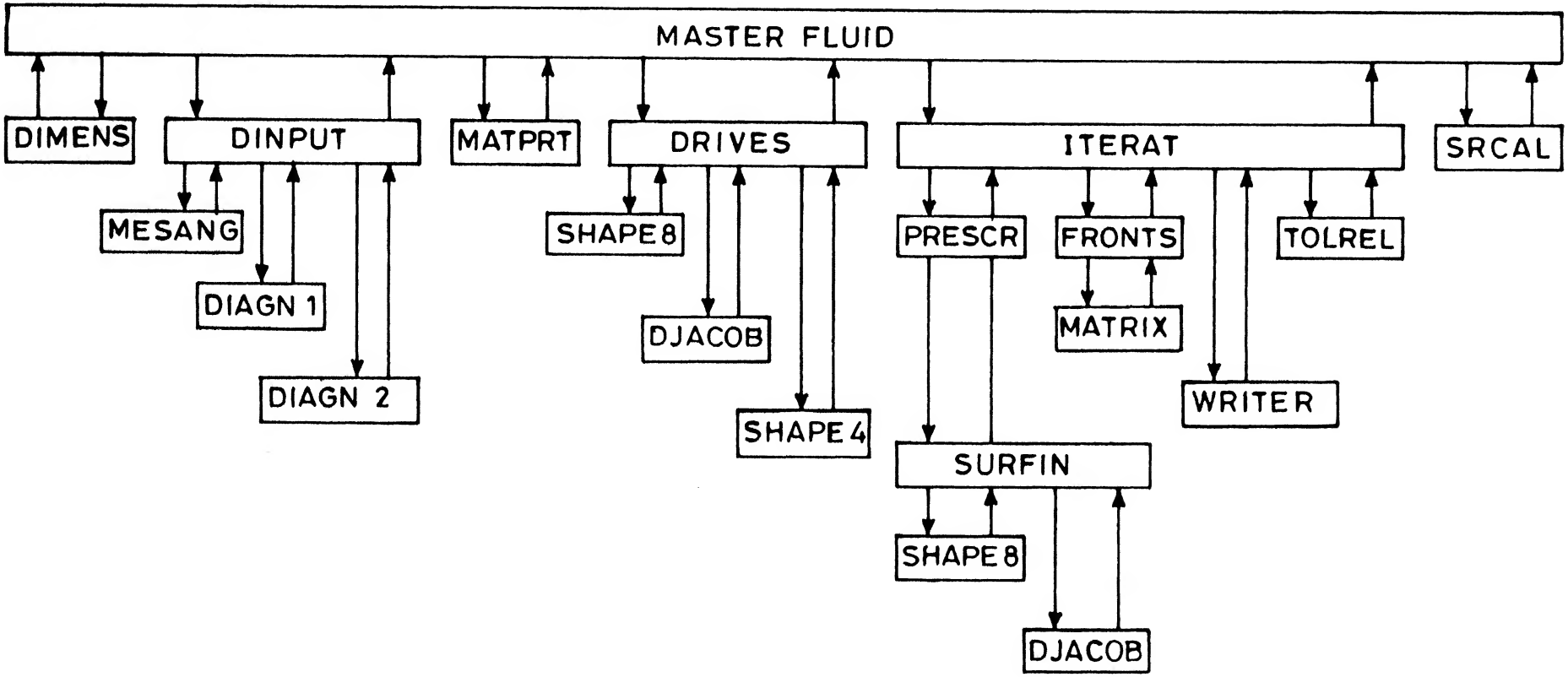


Fig. 3.5 General program structure.

The overall Frontal solution technique covers the following steps :

(i) formation of the element matrices

(ii) assembling into a small, temporary global matrix of chosen size

(iii) introduction of known variable boundary condition

(iv) reduction of the global matrix using Gaussian elimination procedure

(v) back substitution.

The primary objective of the Frontal method is the elimination of variables as soon as possible after their introduction, via appropriate equations, into global matrix. When the contributions from all the elements to a particular node point have been assembled, the corresponding variables associated with that node can be eliminated. The complete matrix is therefore never assembled since the reduced equation can be eliminated from the core and stored on disc. The equations held in core, with the corresponding nodes and variables, are termed as the FRONT and the number of unknown variables in the front is termed as the FRONT WIDTH.

For a non-symmetric global matrix, a preassigned global matrix core area is filled from contributing elements, the largest diagonal entry is the pre-assigned core is then found and used as the pivot in a direct Gaussian-elimination process. As the maximum predetermined number of equations are eliminated, the corresponding reduced equations are written on to disc and more elements and corresponding equations are taken into core.

The equations, nodes and variables currently in core are termed active, those assigned to disc as deactivated and those yet to appear in core as inactive. This is shown diagrammatically in Fig.(3.6).

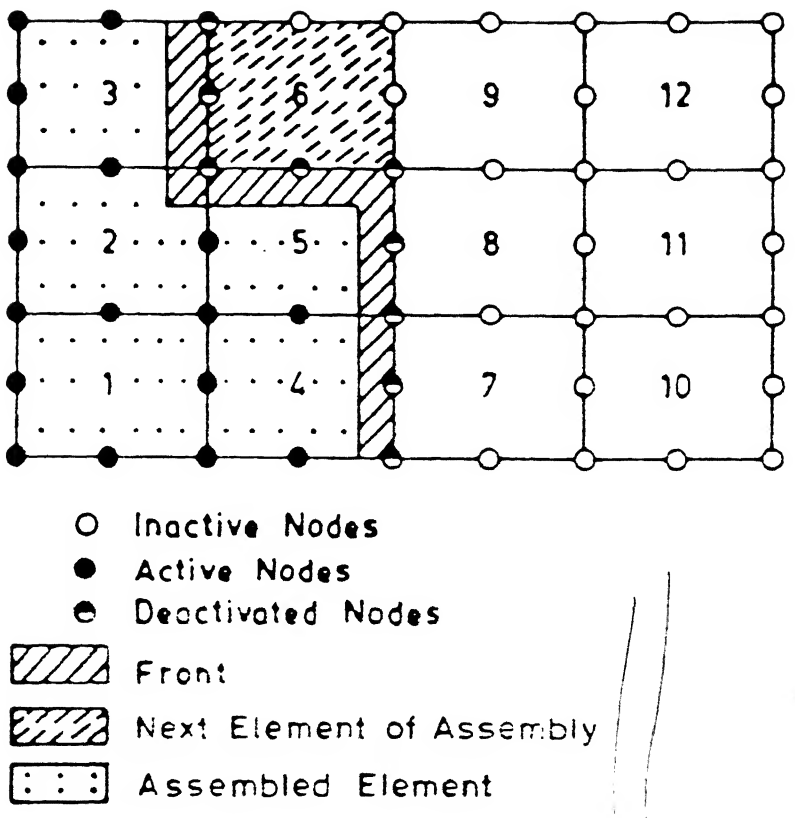


Fig. 3.6 Basic idea of Frontal method.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

The high temperature plastic deformation behaviour of two-phase alloys having the second phase with plate morphology has been modelled. Constitutive equation for high temperature plastic deformation has been taken to be that of a viscoplastic material, i.e. the flow stress is related to strain rate $\dot{\varepsilon}$ by equation $\bar{\sigma} = K \bar{\dot{\varepsilon}}^{m'}$ and pseudo viscosity of the material μ is given as $\mu = K \bar{\dot{\varepsilon}}^{m'-1}$, where K and m' are material constants. The variable parameters of the model were K_2/K_1, the aspect ratio, volume fraction and plate orientation of the second phase. Results of the model calculations have been generated using Finite Element Method.

In this chapter the effect of various parameters during high temperature plastic deformation of two-phase alloys having plate like second phase particles on the averaged values of deviatoric stress components, $\bar{\sigma}'_{11}$ and $\bar{\sigma}'_{22}$ has been discussed. Also trends of stress and strain rate fields around the second phase plate during hot deformation has been examined. The application of results of the present model to high temperature plastic deformation of two-phase structure are also discussed.

4.2 Variation of Stress and Strain Components Over the Two Phase Domain

The variation of σ_{eq}, σ'_{22}, hydrostatic stress P, σ'_{eq},

$\dot{\epsilon}_{22}$ and $\dot{\epsilon}_{eq}$ over the two-phase domain as computed as per section 3.5 of chapter 3 is shown in Figs .4.1 to 4.9. The figures are of two types i.e., 3-D plots and isolines.

Fig.(4.1) shows the variation of σ_{eq} over the two-phase domain for K_2/K_1 = 4.0, i.e. the matrix is considerably softer than the second phase. L/D ratio used for generating this figure was kept as 1.0 with the volume fraction of the second phase kept as 0.24 and slope as 0.0. It can be seen that the stress distribution within each phase is somewhat heterogeneous but the extent of heterogeneity is not very high. While this minor heterogeneity for the softer phase is confined to the interface of two-phases, the heterogeneity in harder phase appears to scan through a much wider area fraction. Further, σ_{eq} is seen to be rising in a very steep way in the harder phase.

Figs. 4.2a and 4.2b show isoline and 3-D plot respectively of the variation of the deviatoric stress component σ'_{22} over the two-phase domain. Similar to Fig.(4.1) average stress is much higher for second phase plate. Also σ'_{22} is seen to be rising in a very steep way in the harder phase. Heterogeneity within each phase is more than that observed for σ_{eq}. Heterogeneity within the phases is in general more in harder phase than in softer matrix and it is maximum close to the two-phase boundary. The variation of σ'_{22}, close to the phase boundaries as clearly evident from Fig.(4.2b) is somewhat sinusoidal.

Figs. 4.3a and 4.3b show isoline and 3-D plot respectively of the variation of the hydrostatic stress.

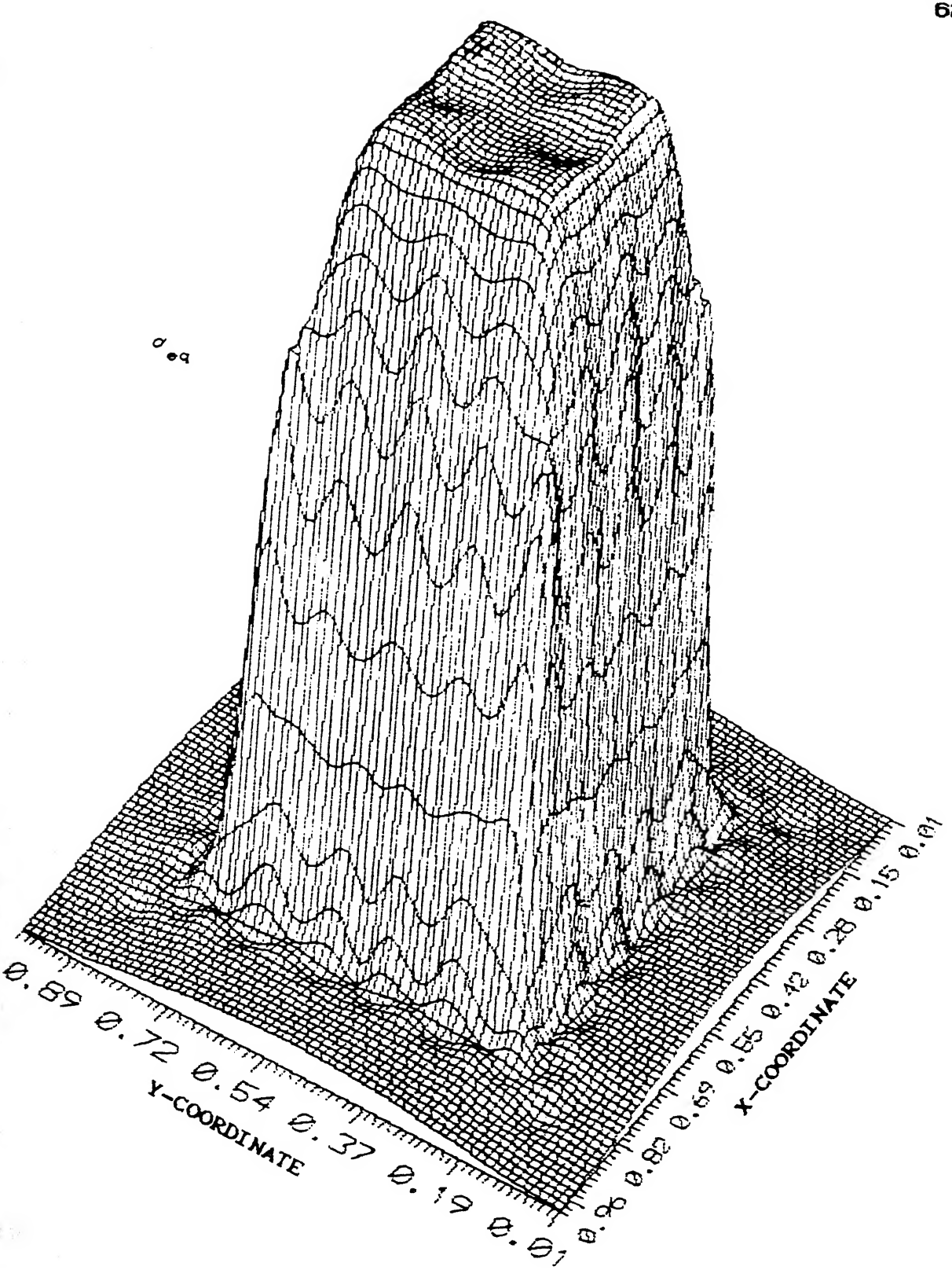


Fig.4.1 A 3-D plot of instantaneous σ_{eq} values over the domain (K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

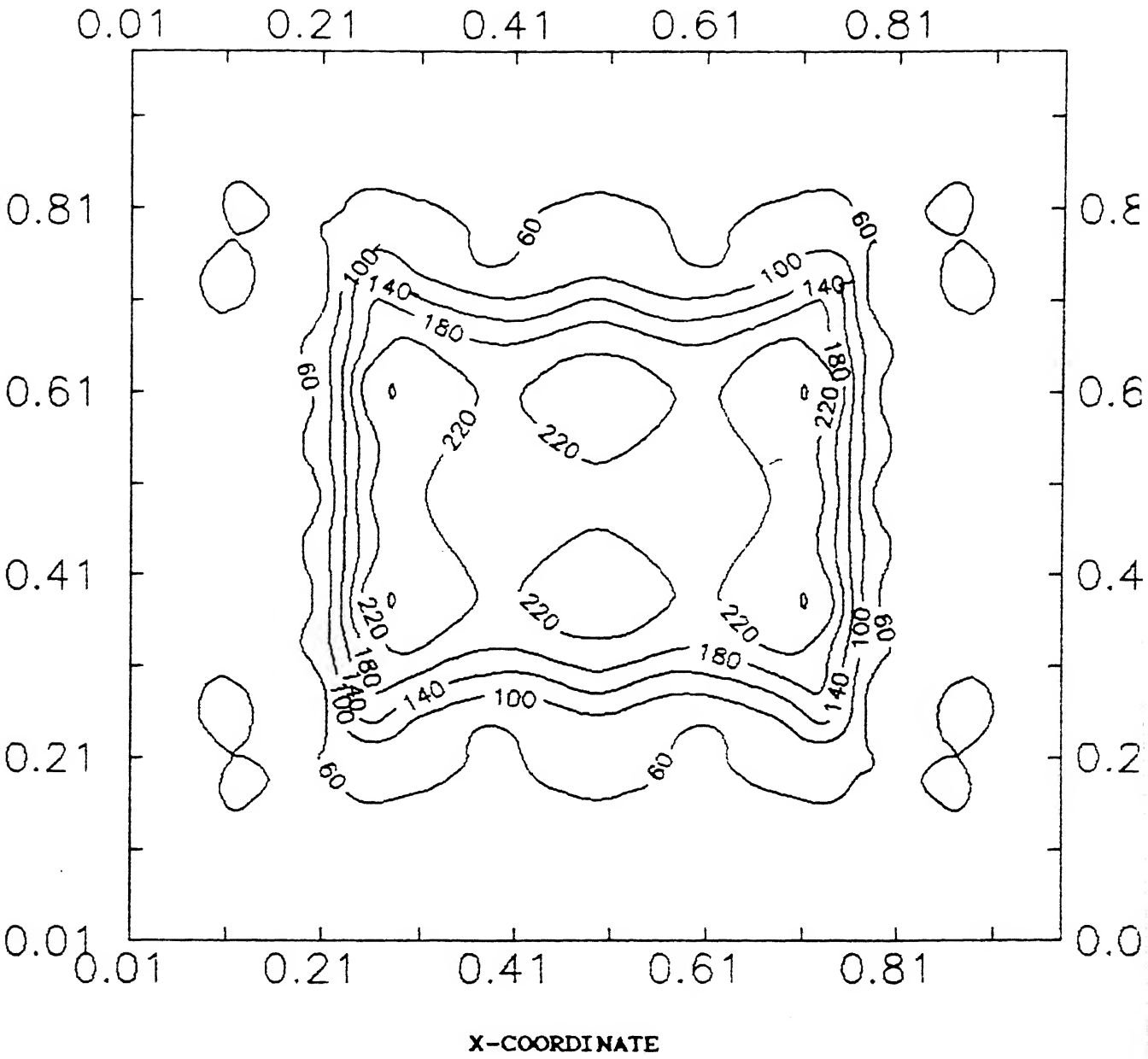


Fig.4.2a Isolines for instantaneous σ'_{22} values over the domain

(K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

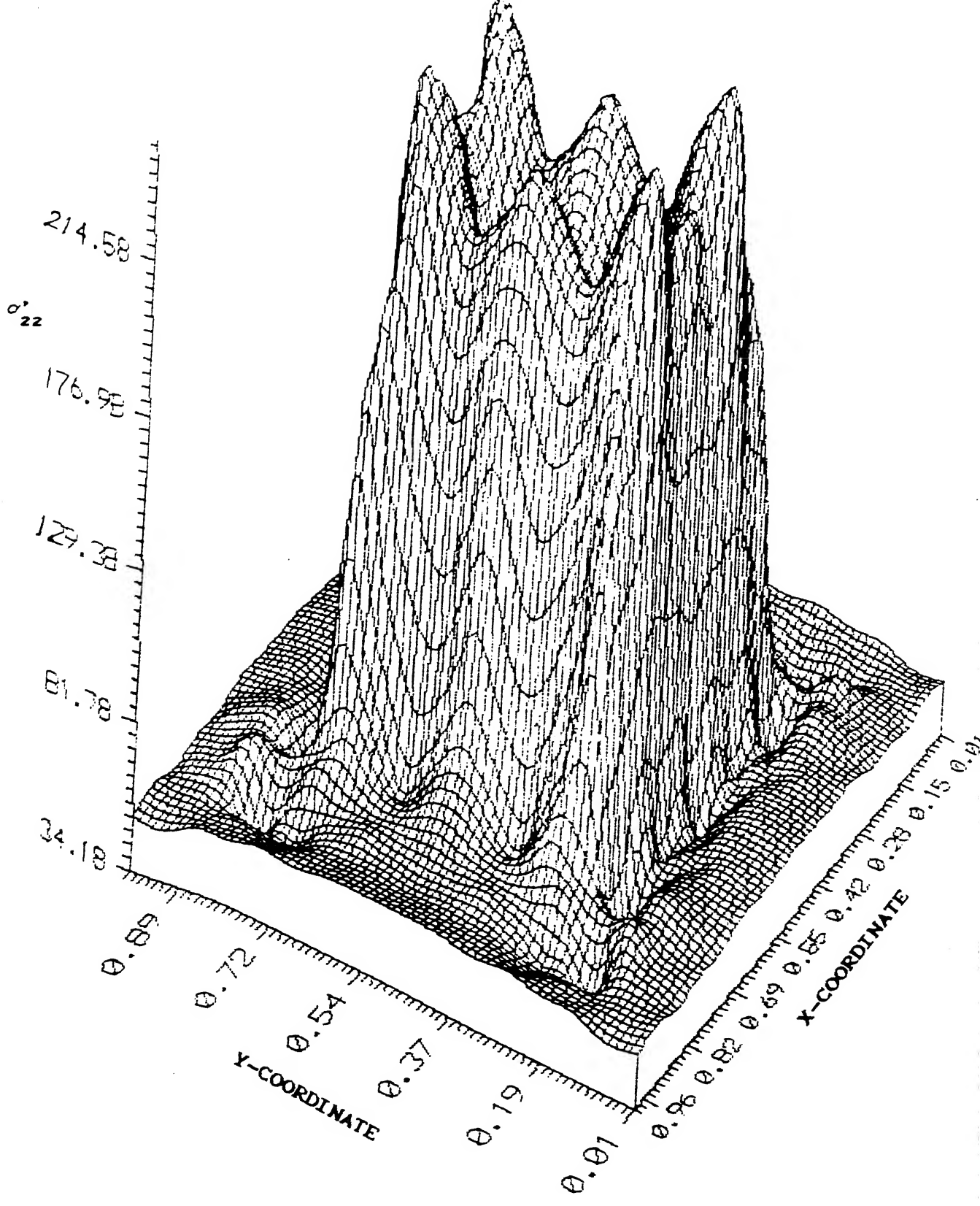


Fig.4.2b A 3-D plot of instantaneous σ'_{22} values over the domain (K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

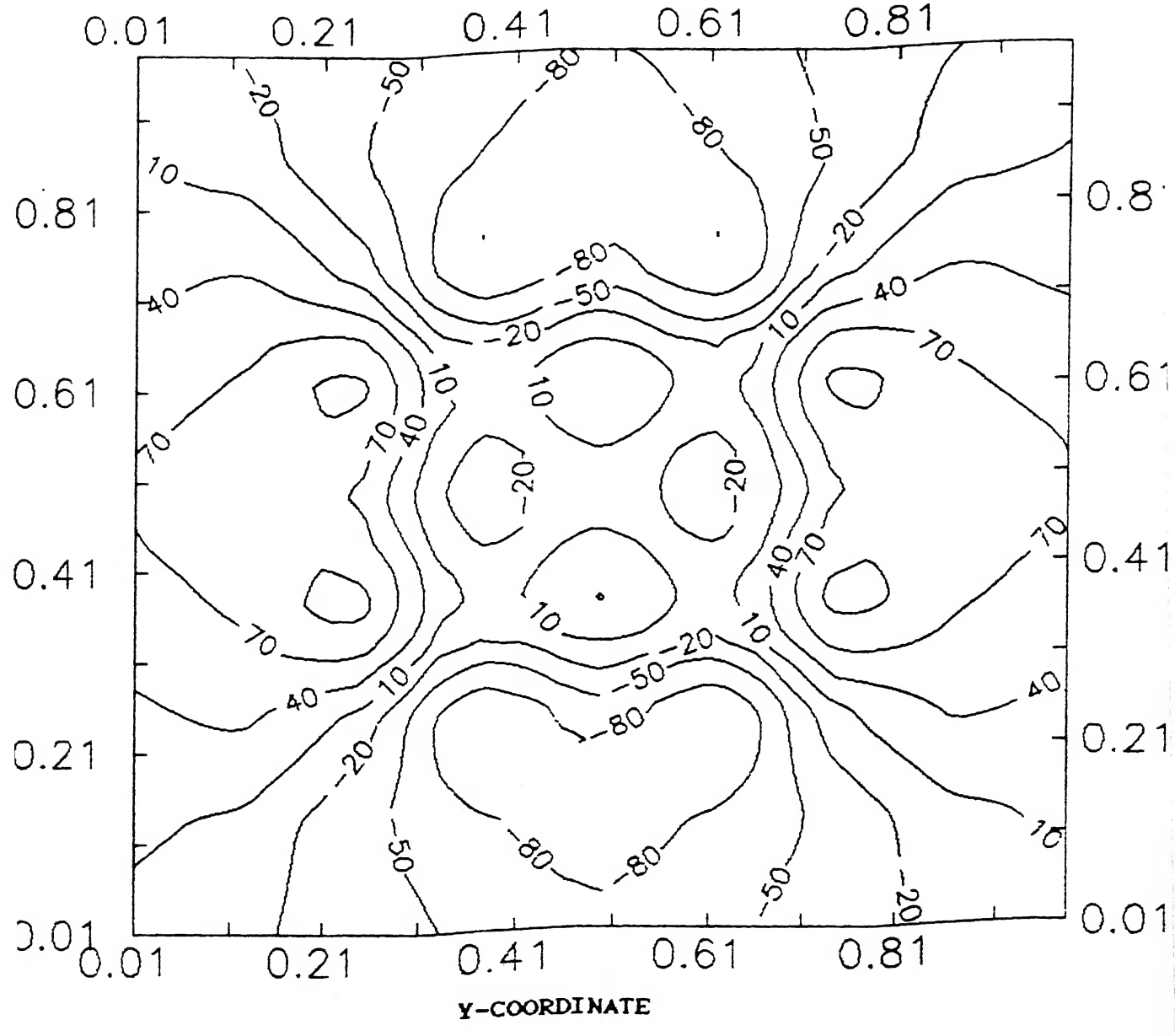


Fig.4.3a Isolines for instantaneous β values over the domain (K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

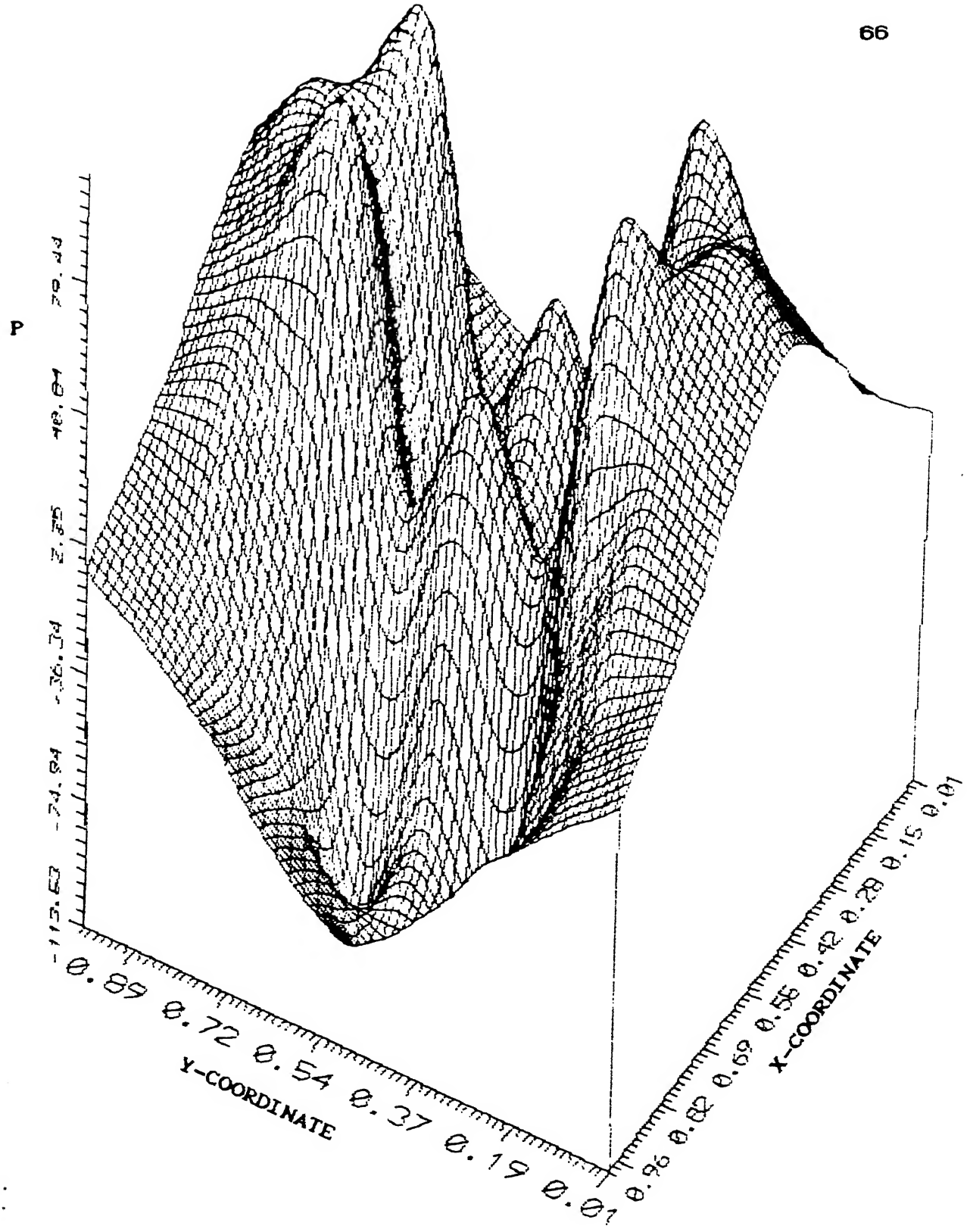


Fig.4.3b A 3-D plot of instantaneous P values over the domain

(K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

Heterogeneity in this is more than in σ'_{22} and σ'_{ef}. Here also close to the two-phase boundary the heterogeneity is maximum. Fig. 4.4 shows the variation of σ'_{eq} over the two-phase domain. The nature of the plot is similar to that of σ_{eq}.

Figs. 4.5a and 4.5b show isoline and 3-D plot respectively of the variation of the $\dot{\varepsilon}_{22}$. Heterogeneity in general is much more than stress fields. Heterogeneity in the harder second phase is much less than that observed in the matrix phase. Total eight high peaks can be observed in the matrix phase. Four of them lie on the mid-points of the domain edges. Figs. 4.6a and 4.6b show isoline and 3-D plot for $\dot{\varepsilon}_{ef}$. The nature is same as that of $\dot{\varepsilon}_{22}$. Only a bit more heterogeneity within second phase is observed.

Fig. 4.7 is the isoline of σ'_{eq} for a domain with $K_2/K_1 = 1.5$, $L/D = 2.0$, $V_f = 0.44$ and plate orientation at 45^{o}. It is evident from this that the nature of plot for σ'_{eq} is not affected by these model parameters. Figs. 4.8 and 4.9 are the isolines of $\dot{\varepsilon}_{22}$ and $\dot{\varepsilon}_{ef}$ for this domain. When compared with Figs. 4.5 and 4.6 we find not much difference in nature.

4.3 The Effect of Various Parameters of the Model on $\bar{\sigma}'_{11}$ and $\bar{\sigma}'_{22}$

4.3.1 Effect of change of boundary conditions

The averaged stresses, as computed as per section 3.5, for a given two-phase structure (K_2/K_1, V_f, L/D and m = const.) should be invariant to the flow field or the boundary condition.

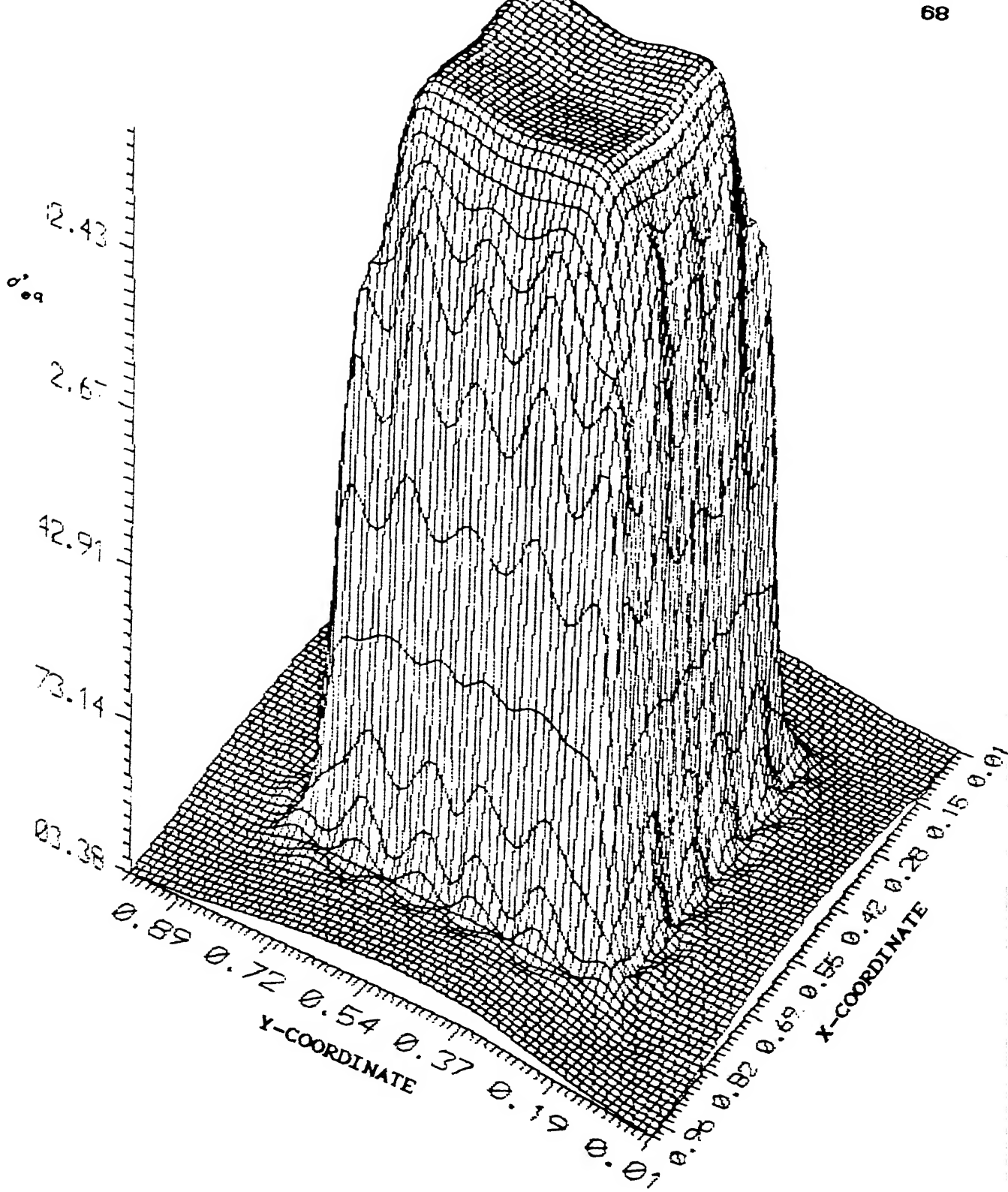


Fig.4.4 A 3-D plot of instantaneous σ'_{eq} values over the domain (K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

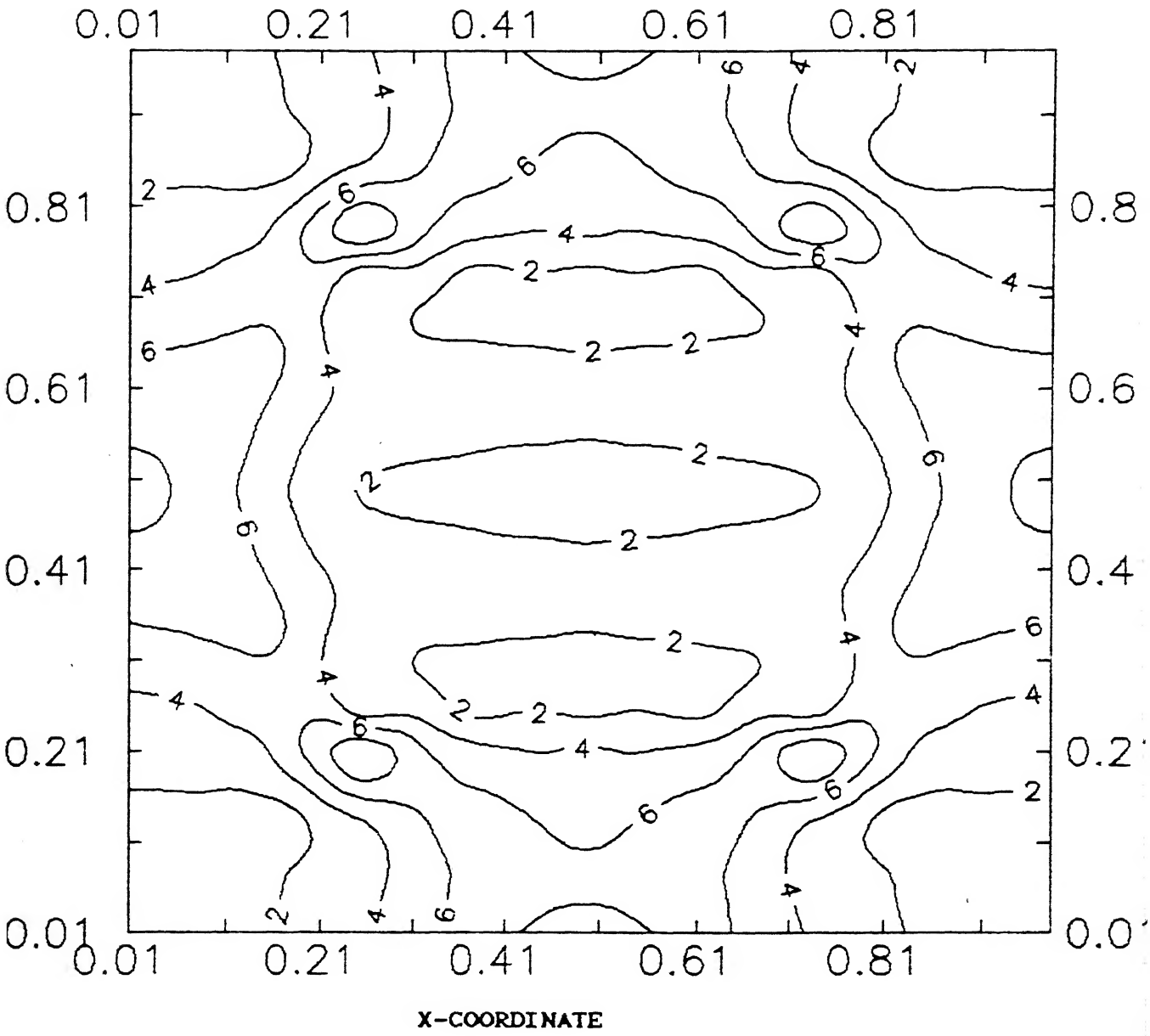


Fig.4.5a Isolines for instantaneous $\dot{\varepsilon}_{22}$ values over the domain

(K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

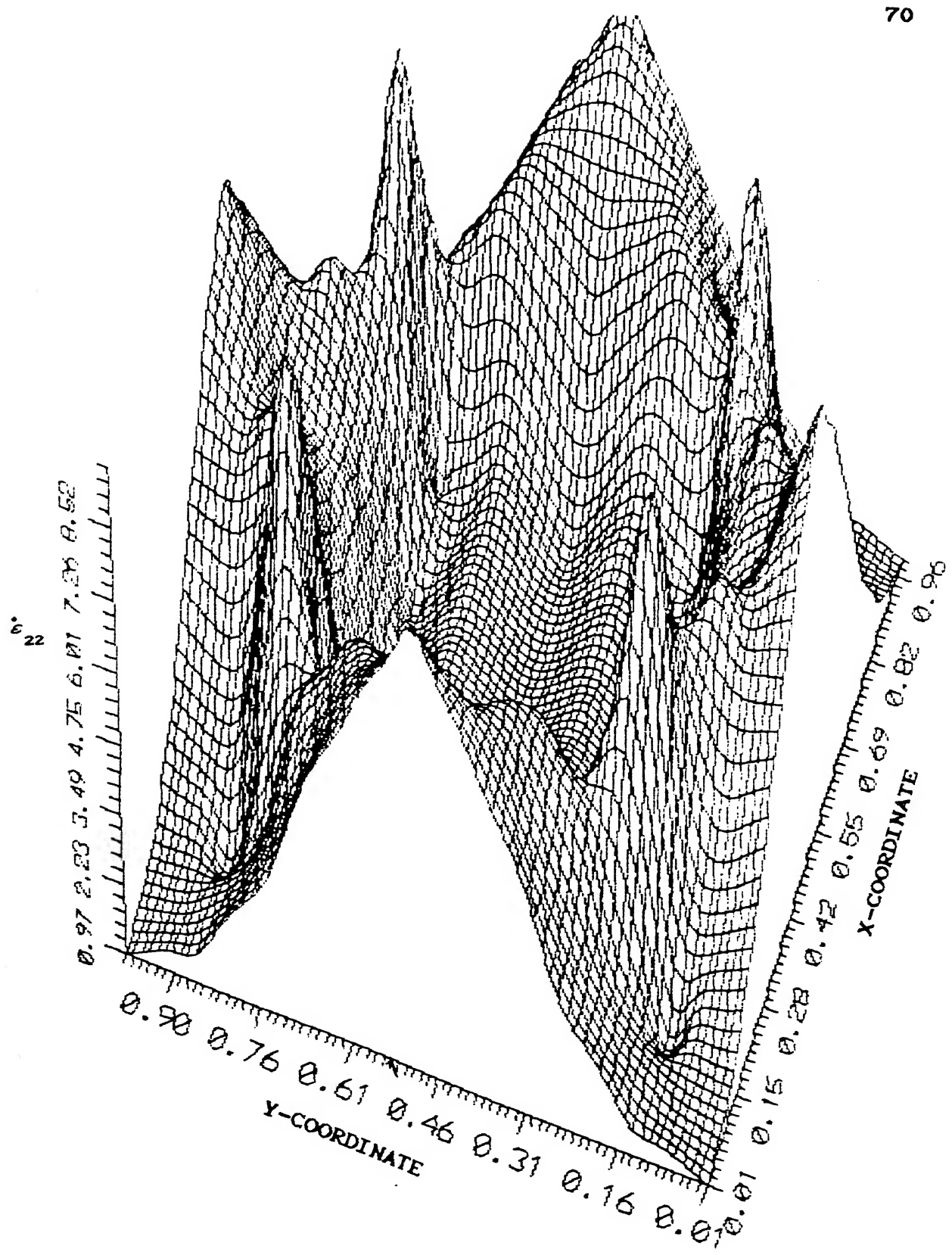


Fig.4.5b A 3-D plot of instantaneous $\dot{\varepsilon}_{22}$ values over the domain

(K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

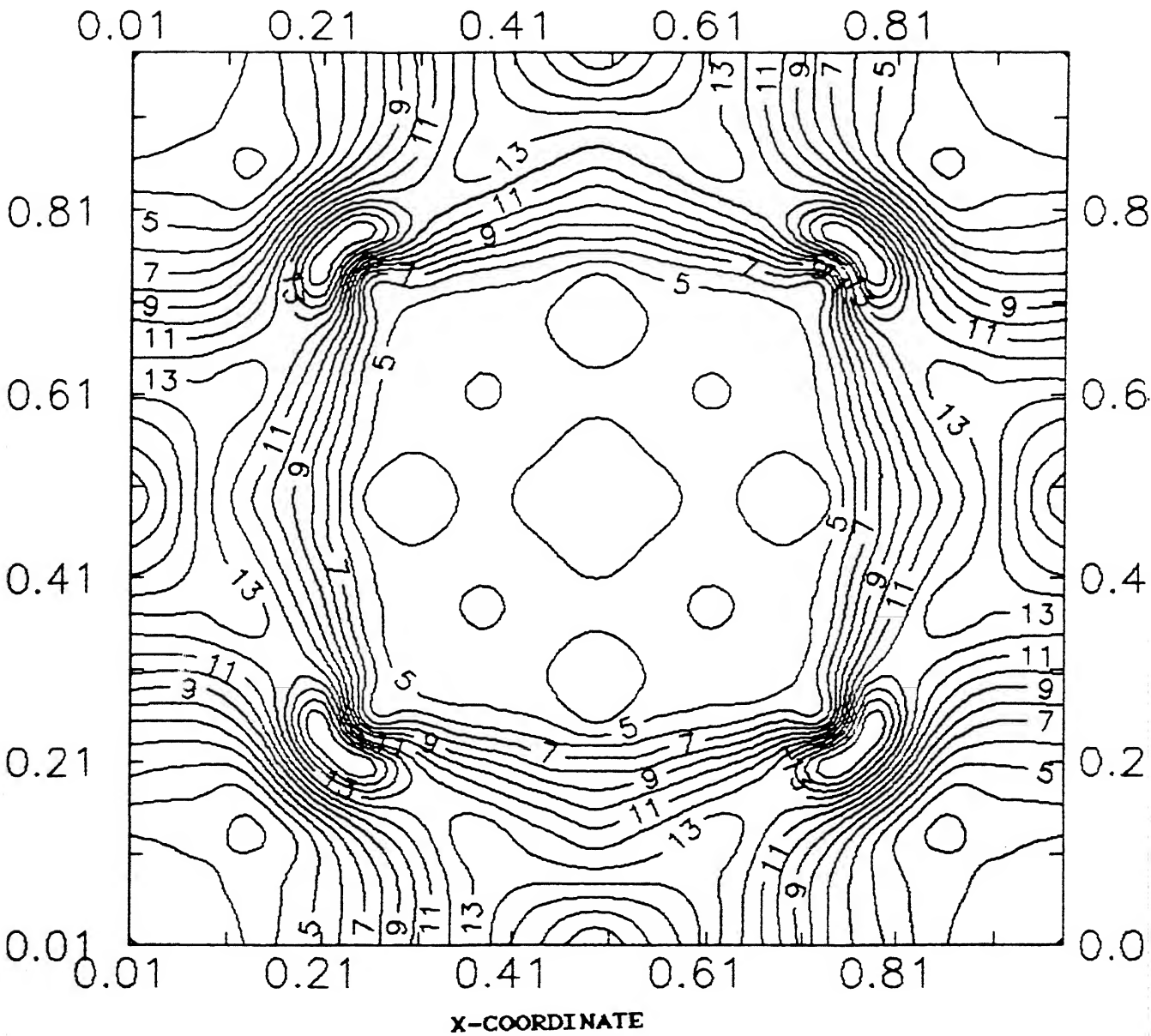


Fig.4.6a Isolines for instantaneous $\dot{\varepsilon}_{eq}$ values over the domain

(K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

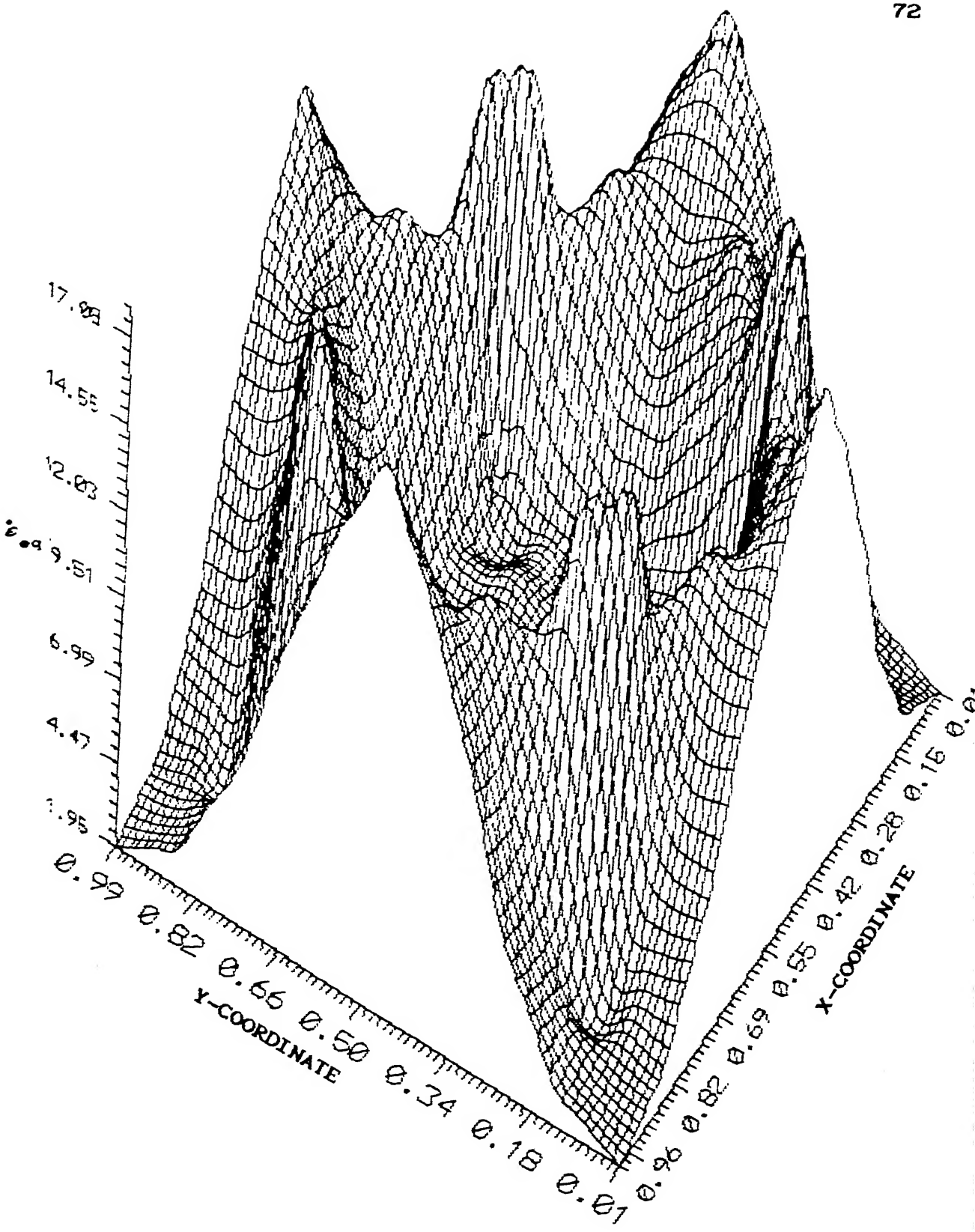


Fig.4.6b A 3-D plot of instantaneous $\dot{\varepsilon}_{eq}$ values over the domain

(K_2/K_1 = 4.0, L/D = 1.0, V_f = 0.24, m = 0)

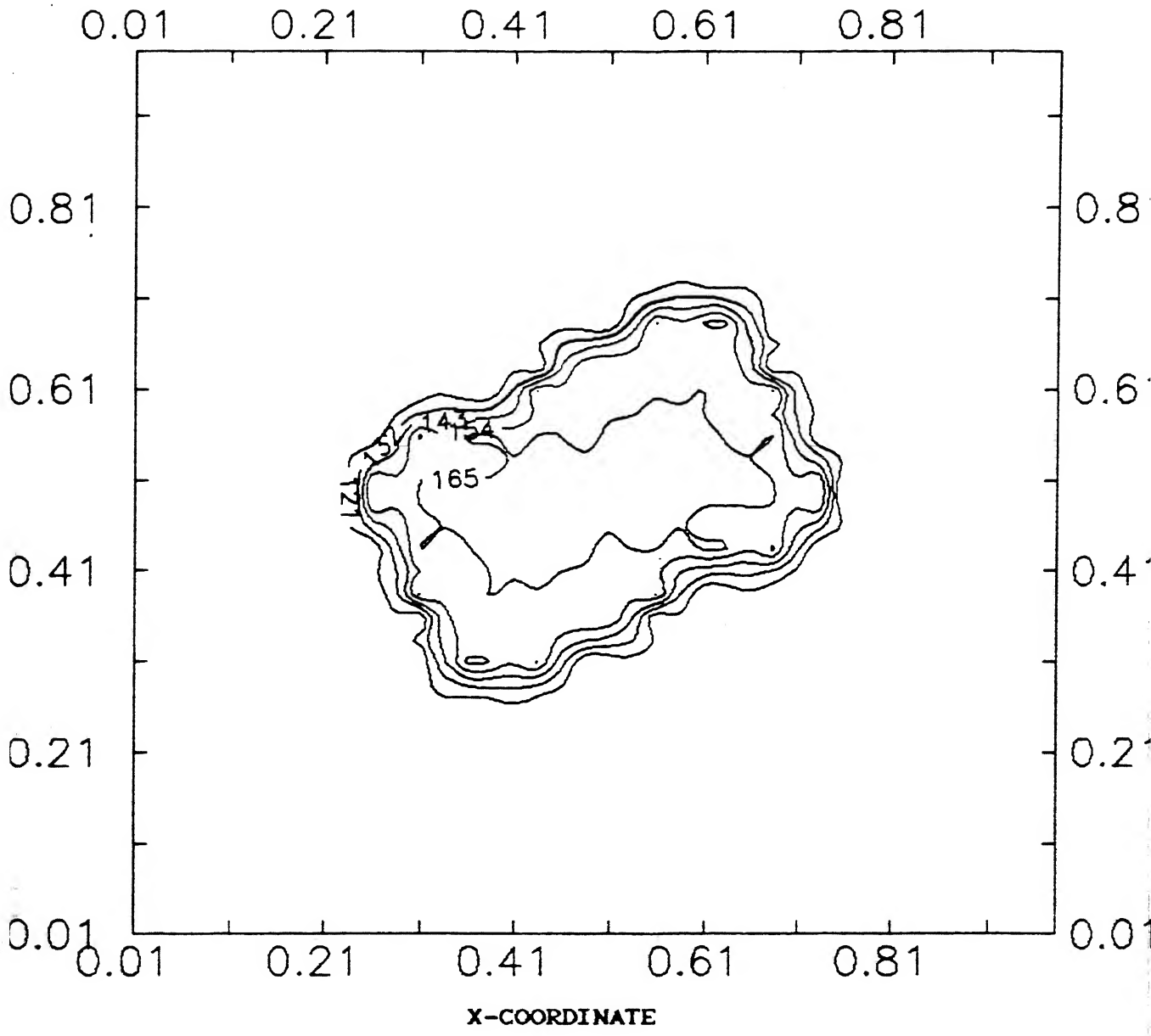


Fig.4.7 Isolines for instantaneous σ'_{eq} values over the domain (K_2/K_1 = 1.5, L/D = 2.0, V_f = 0.44, m = 0.5)

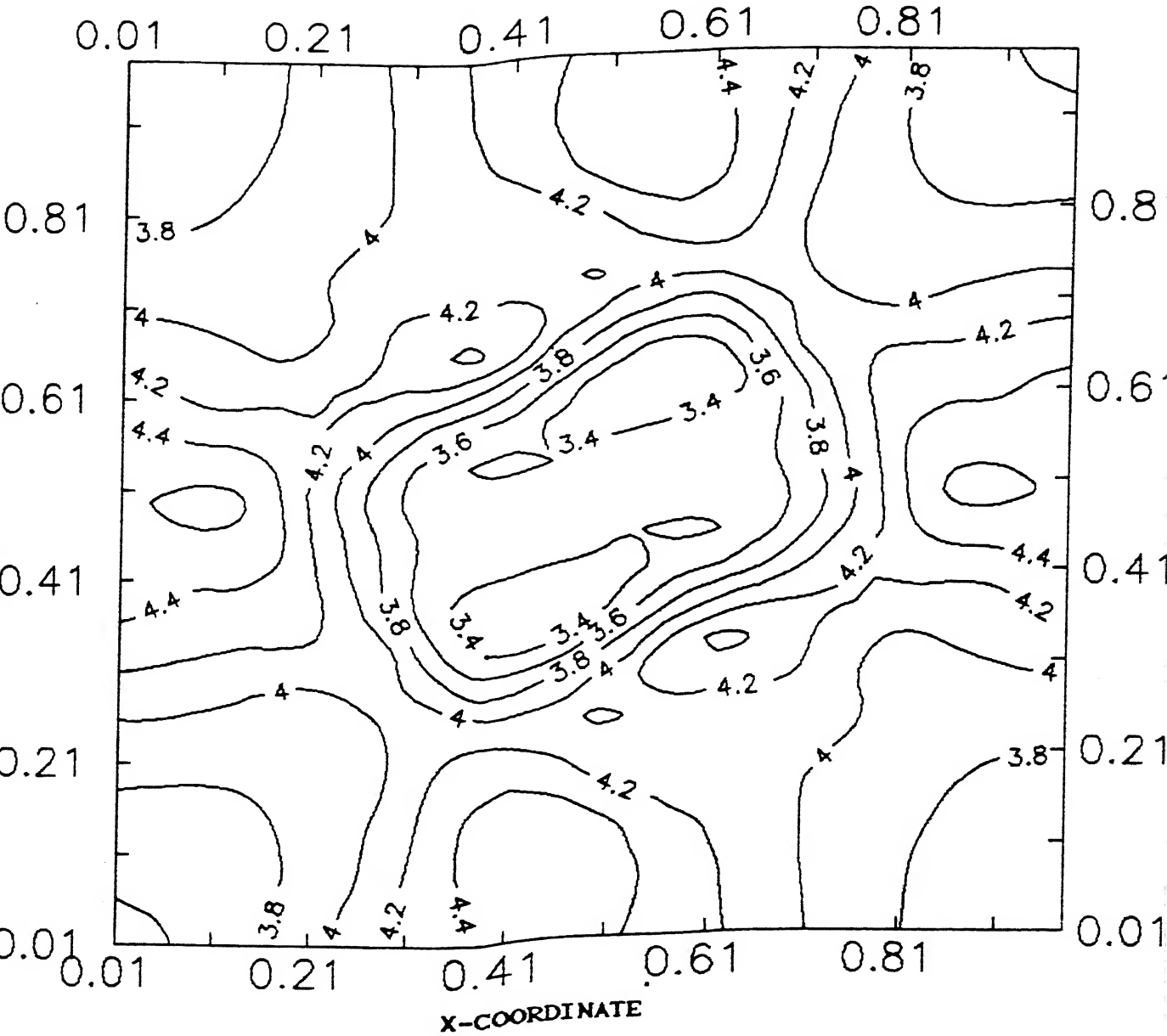


Fig.4.8 Isolines for instantaneous $\dot{\varepsilon}_{zz}$ values over the domain (K_2/K_1 = 1.5, L/D = 2.0, V_f = 0.44, m = 0.5)

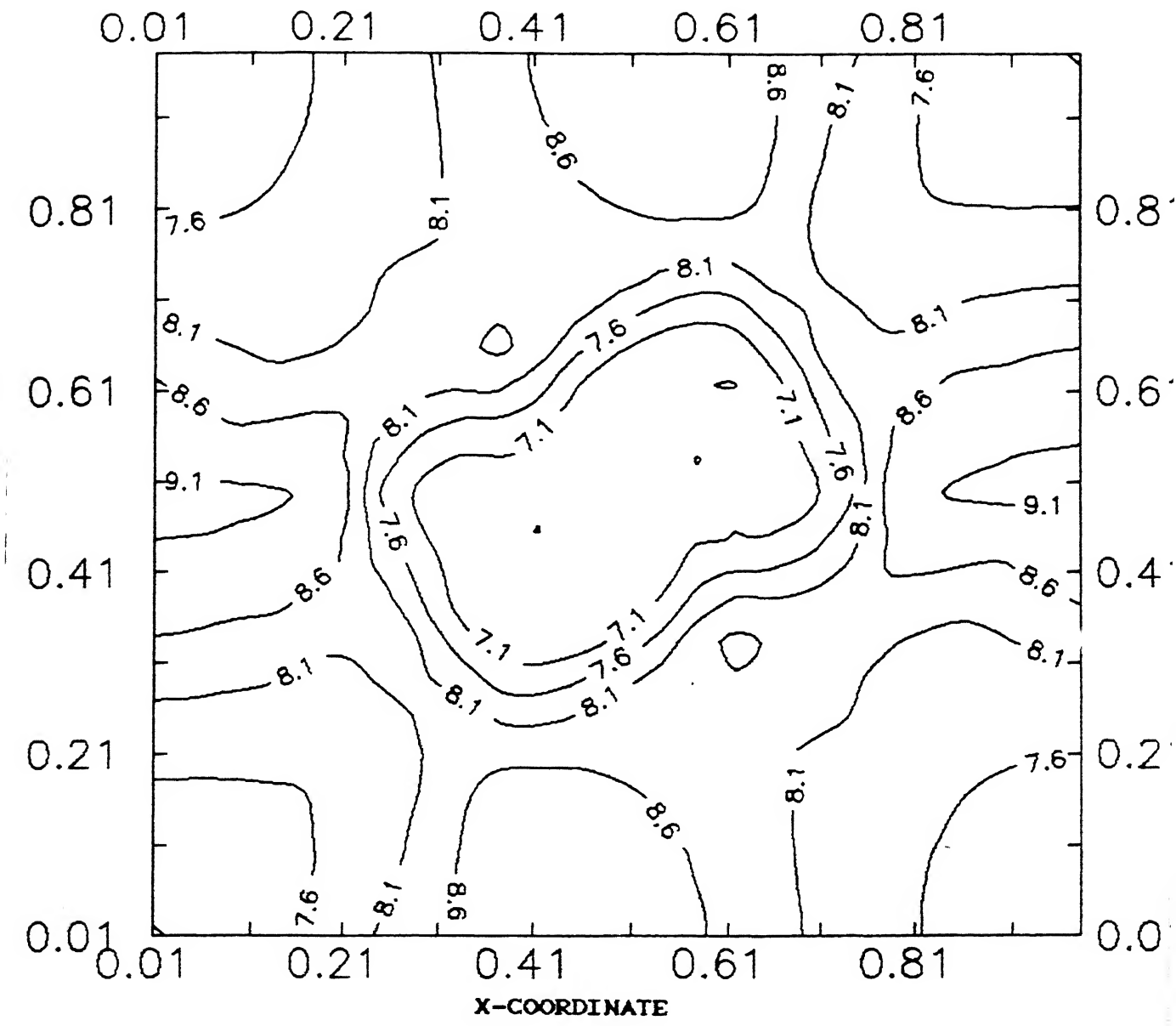


Fig.4.9 Isolines for instantaneous $\dot{\varepsilon}_{ef}$ values over the domain

(K_2/K_1 = 1.5, L/D = 2.0, V_f = 0.44, m = 0.5)

Different boundary conditions (as given in appendix) were applied to domains with different sets of m, V_f, L/D values for K_2/K_1 = 4.0. The change in the value of effective pseudo viscosity was studied. Table 4.1 gives some representative data. Variation in $-\bar{\sigma}'_{11}$ and $\bar{\sigma}'_{22}$ with change in boundary conditions which give different value of average shear strain rates, is very small. And μ_{eff} calculated using these values and $\dot{\varepsilon}_{11}$ and $\dot{\varepsilon}_{22}$ value show very little (~ 3%) variation. Thus it can be inferred that μ_{eff} calculated in this way is invariant to stress and strain rate fields. Another interesting point to be noted from the table is that $\bar{\sigma}'_{eq}$ is nearly same for all the datas. μ_{eff} as calculated using $\bar{\sigma}'_{eff}$ and $\dot{\varepsilon}_{eff}$ cannot be said to be invariant of flow field.

4.3.2 Effect of K_2/K_1

Effect of K_2/K_1 as shown in Figs. 4.10 to 4.17 which is varied from two to five has been studied. The variation of averaged deviatoric stress components $-\bar{\sigma}'_{11}$ and $\bar{\sigma}'_{22}$ with K_2/K_1 is almost linear. This is true for any value of m, V_f and L/D. This fact indicates that for any morphology, the averaged stress-components for any two-phase can be calculated if the stress-component averages are known for two values of K_2/K_1. Anisotropy in general increases with K_2/K_1. It is substantial when V_f is low and L/D is on higher side and m is not close to 0.5.

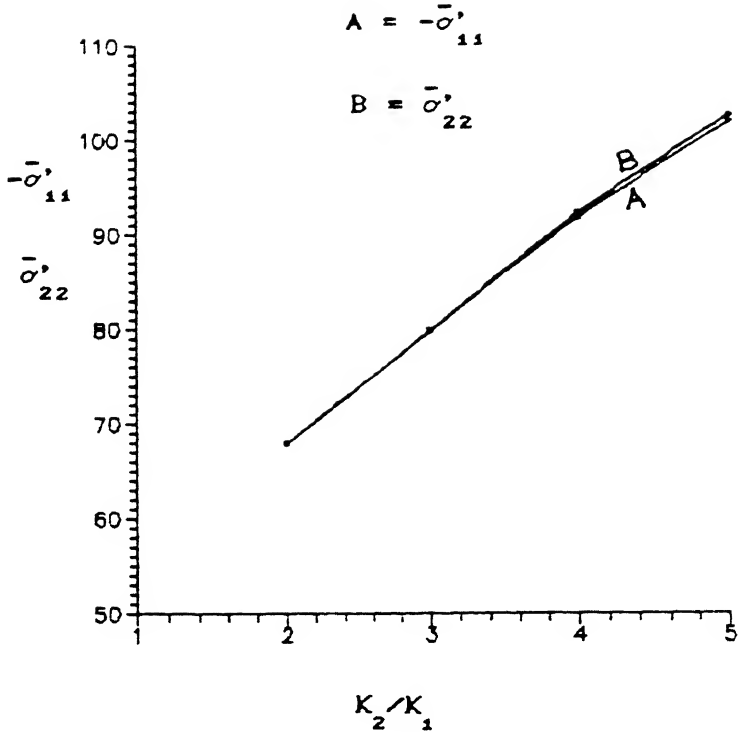


Fig.4.10 Variation of averaged stress components with K_2/K_1 (V_f = 0.24, L/D = 1.5, m = 7.0)

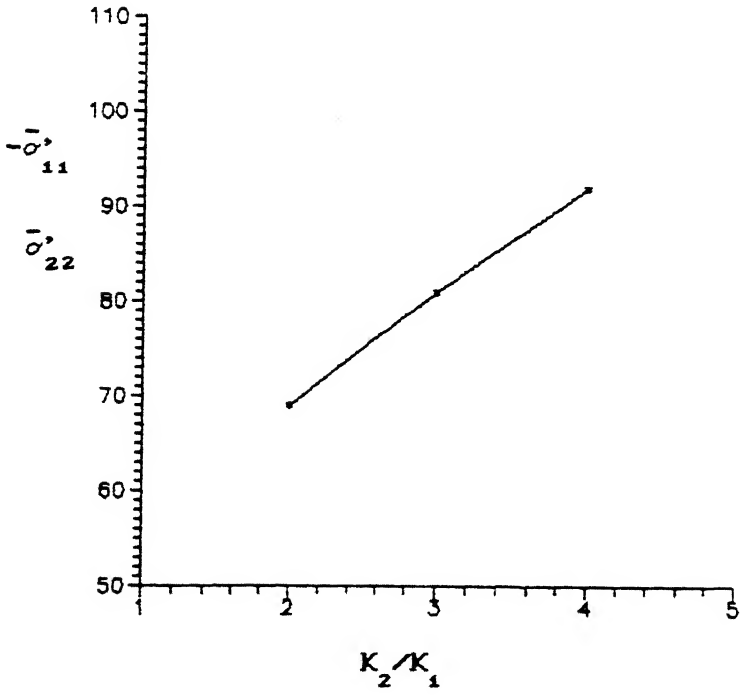


Fig.4.11 Variation of averaged stress components with K_2/K_1 (V_f = 0.24, L/D = 1.5, m = 0.5)

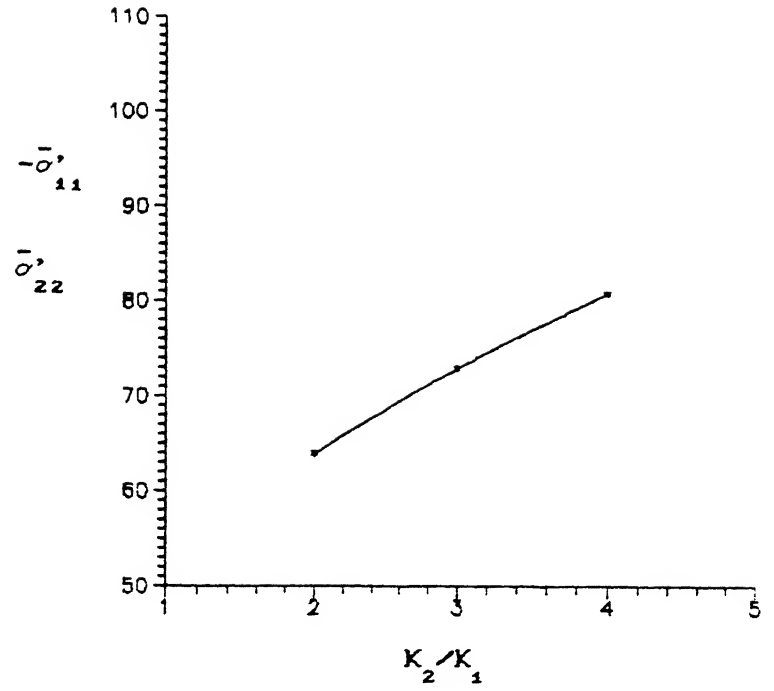


Fig.4.12 Variation of averaged stress components with K_2/K_1 (V_f = 0.16, L/D = 4.0, m = 0.5)

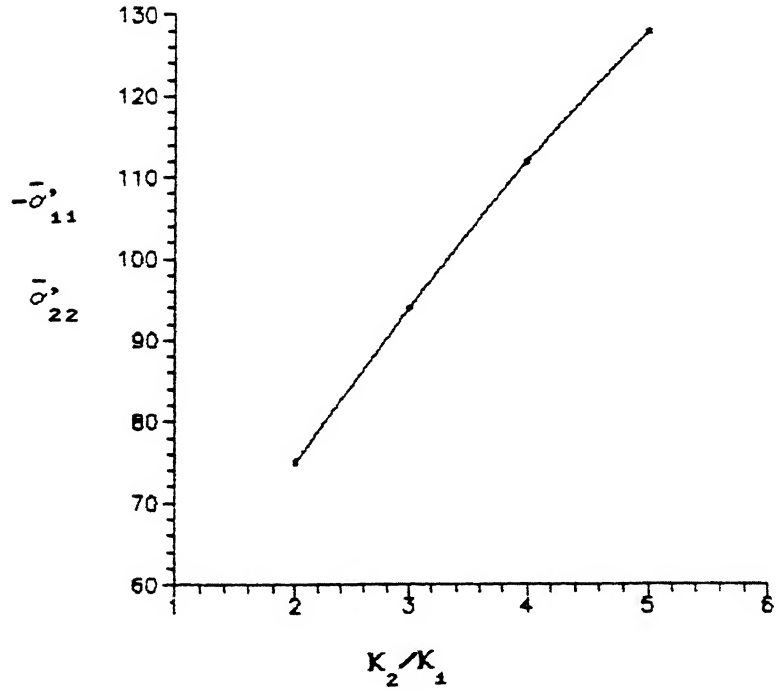


Fig.4.13 Variation of averaged stress components with K_2/K_1 (V_f = 0.36, L/D = 1.0, m = 1.5)

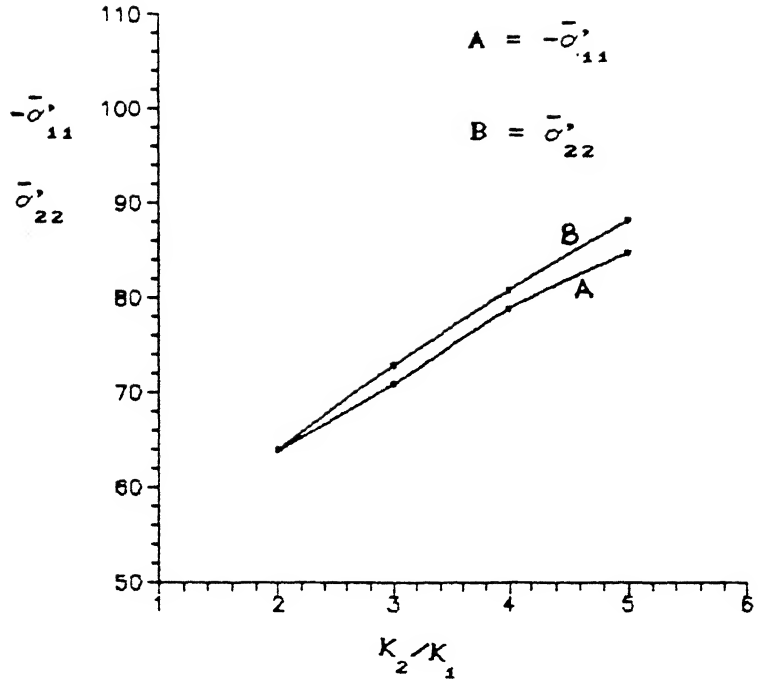


Fig.4.14 Variation of averaged stress components with K_2/K_1 (V_f = 0.16, L/D = 4.0, m = 1.5)

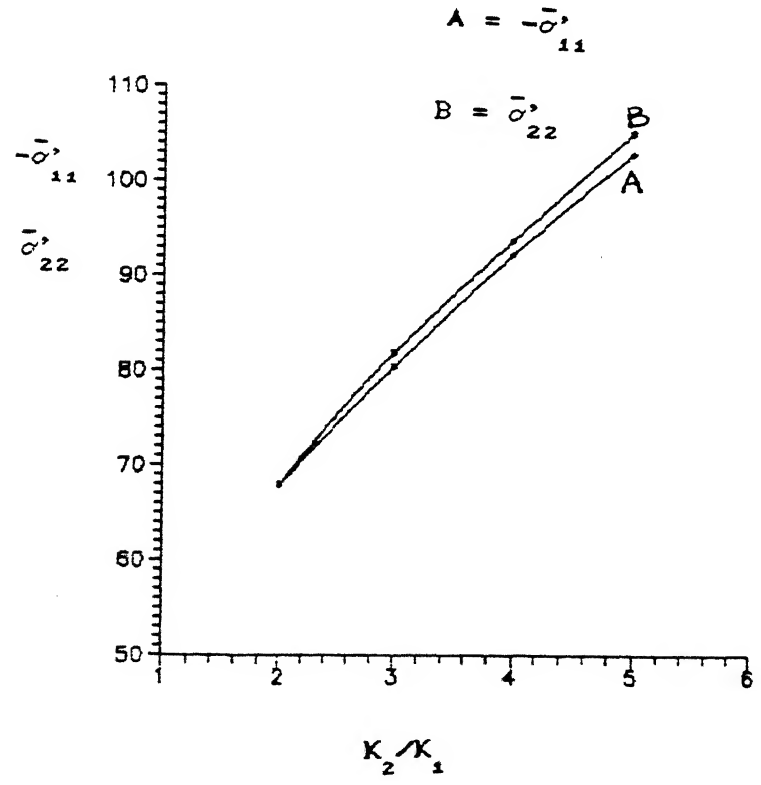


Fig.4.15 Variation of averaged stress components with K_2/K_1 (V_f = 0.24, L/D = 1.5, m = 5.0)

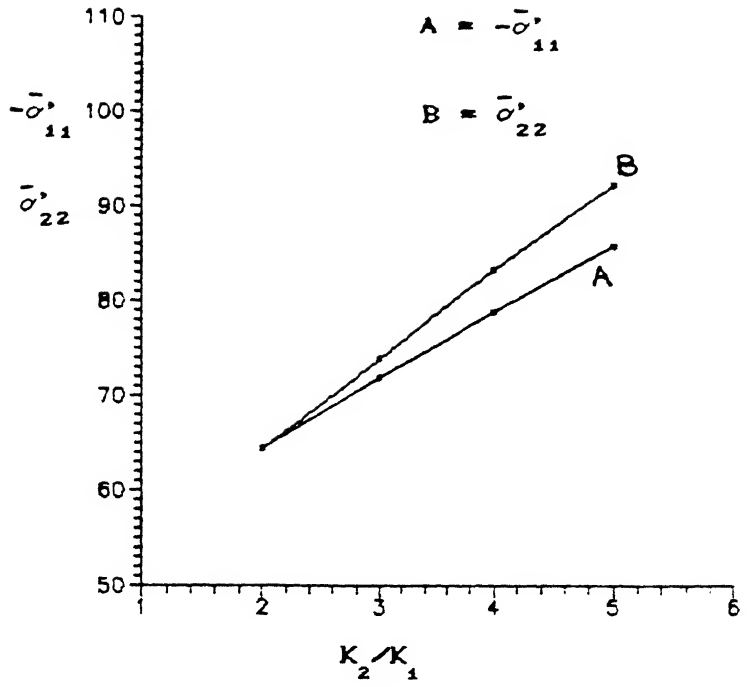


Fig.4.16 Variation of averaged stress components with K_2/K_1 (V_f = 0.16, L/D = 4.0, m = 5.0)

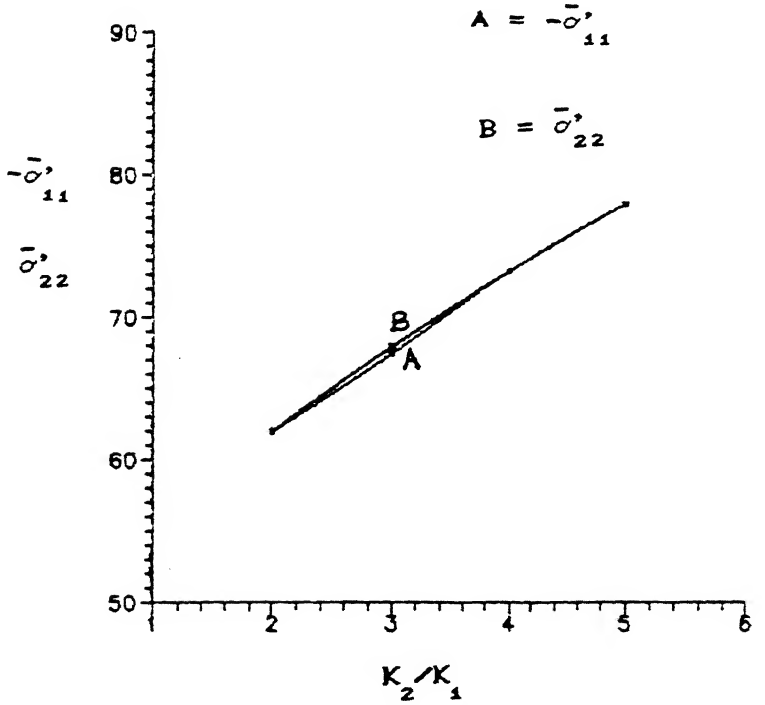


Fig.4.17 Variation of averaged stress components with K_2/K_1 (V_f = 0.12, L/D = 3.0, m = 5.0)

TABLE 4.1

Change in Effective μ with Boundary Condition

Sl. No.	$\frac{K_2}{K_1}$	V_f	L/D	m	Boundary condition	$\bar{\sigma}'_{22}$	$\dot{\varepsilon}_{22}$	$\bar{\sigma}'_{eq}$	$\dot{\varepsilon}_{eq}$	$\mu \frac{\bar{\sigma}'_{22}}{\dot{\varepsilon}_{22}}$	$\mu \frac{\bar{\sigma}'_{eq}}{\dot{\varepsilon}_{eq}}$
1	4	0.44	1.5	0.5	BC1 ($\dot{\varepsilon}_{12}$=0)	126.1	4	254.4	8.0	31.5	31.7
2	4	0.44	1.5	0.5	BC2 ($\dot{\varepsilon}_{12}$=0.85)	122.9	4	255.3	'9.2	30.7	27.7
3	4	0.44	1.5	0.5	BC3 ($\dot{\varepsilon}_{12}$=1.2)	122.2	4	255.3	10.6	30.5	24.1
4	4	0.44	2.0	0.5	BC1 ($\dot{\varepsilon}_{12}$=0)	127.5	4	256.8	8.0	31.9	32.0
5	4	0.44	2.0	0.5	BC2 ($\dot{\varepsilon}_{12}$=0.85)	124.2	4	253.1	9.2	31.0	27.5
6	4	0.44	2.0	0.5	BC3 ($\dot{\varepsilon}_{12}$=1.2)	123.6	4	256.5	10.6	30.5	24.2

4.3.3 Effect of volume fraction

As shown in Fig.(4.18) for low values of volume fraction (<0.25), $-\bar{\sigma}'_{11}$ and $\bar{\sigma}'_{22}$ vary almost linearly as a function of volume fraction irrespective of morphology and orientation. For higher values of volume fraction $-\bar{\sigma}'_{11}$ and $\bar{\sigma}'_{22}$

are sensitive to plate L/D and plate orientation. The variation of these quantities with respect to V_f for a given orientation and L/D is shown in Figs. 4.19 and 4.20.

4.3.4 Effect of L/D

The effect of L/D on averaged stress components as stated in section 2.3 is negligible for V_f less than 0.25. For higher volume fraction, as shown in Figs. 4.21 to 4.23, averaged stresses are sensitive to L/D. The nature of variation is oscillating. Anisotropy is much high for high values of L/D except when m is equal to 0.5 (plate is oriented at 45°). This effect is maximum for m = 0 (plate is parallel to domain edges).

4.3.5 Effect of plate orientation

The effect of plate orientation is shown in Figs. 4.24 to 4.27. For $V_f < 0.25$ there is no significant variation of averaged deviatoric stress components with m. For L/D = 1, $-\bar{\sigma}'_{11}$ and $\bar{\sigma}'_{22}$ variation with m is very small for V_f varying from 0.16 to 0.44. And for other values of L/D, $-\bar{\sigma}'_{11}$ curve crosses the $\bar{\sigma}'_{22}$ curve at m = 0.5.

4.4 Applications of the Results of the Model for the Analysis of High Temperature Deformation Processing of Two Phase Alloys

As stated in chapter 2 the two important objectives of the analysis of deformation processing of two-phase alloys are :

(a) transformation of initial microstructure to a

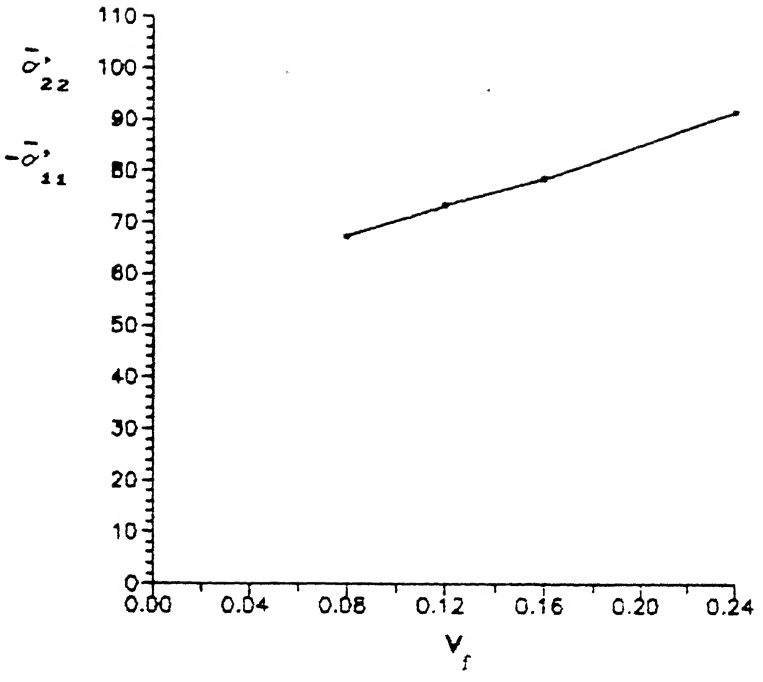


Fig.4.18 Variation of averaged stresses with V_f (for V_f < 0.25)

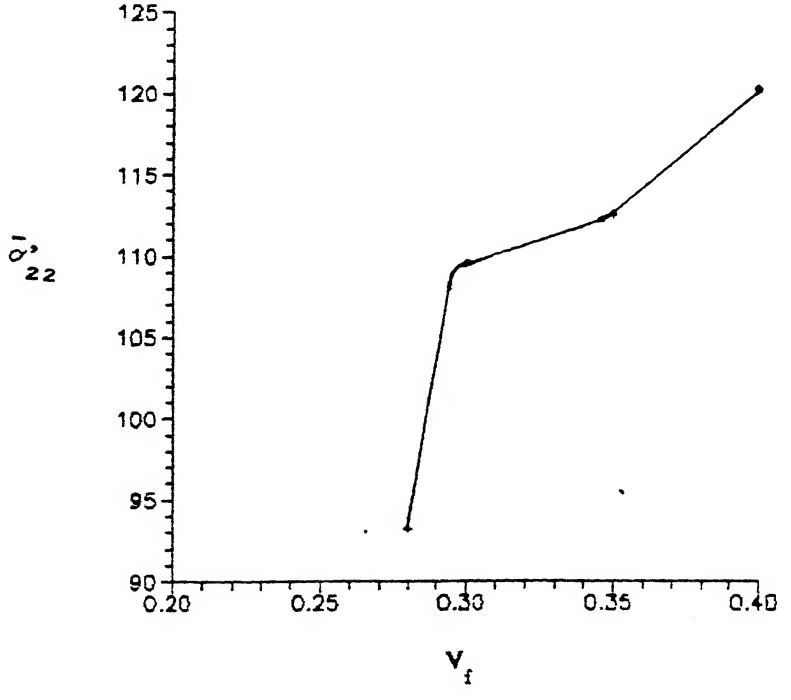


Fig.4.19 Variation of $\bar{\sigma}'_{22}$ with V_f (L/D = 2, K_2/K_1 = 4.0, m = 0)

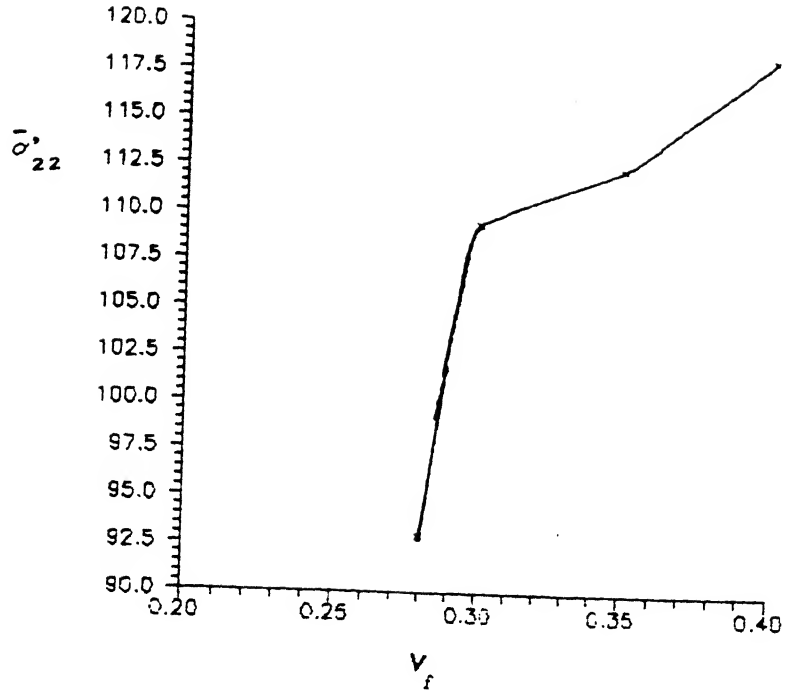


Fig.4.20 Variation of $\bar{\sigma}'_{22}$ with V_f (L/D = 2, K_2/K_1 = 4.0, m = 0.5)

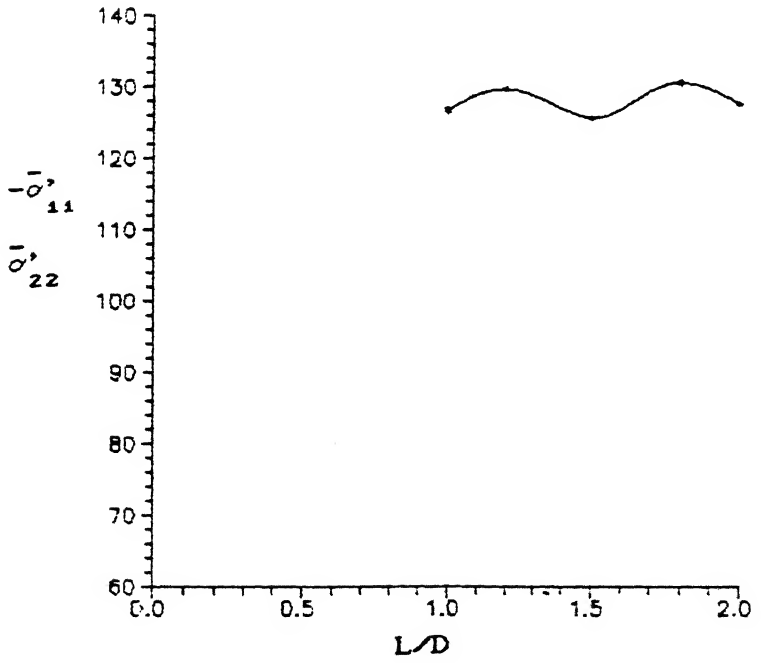


Fig.4.21 Variation of averaged stress components with L/D (V_f = 0.44, K_2/K_1 = 4.0, m = 1.0)

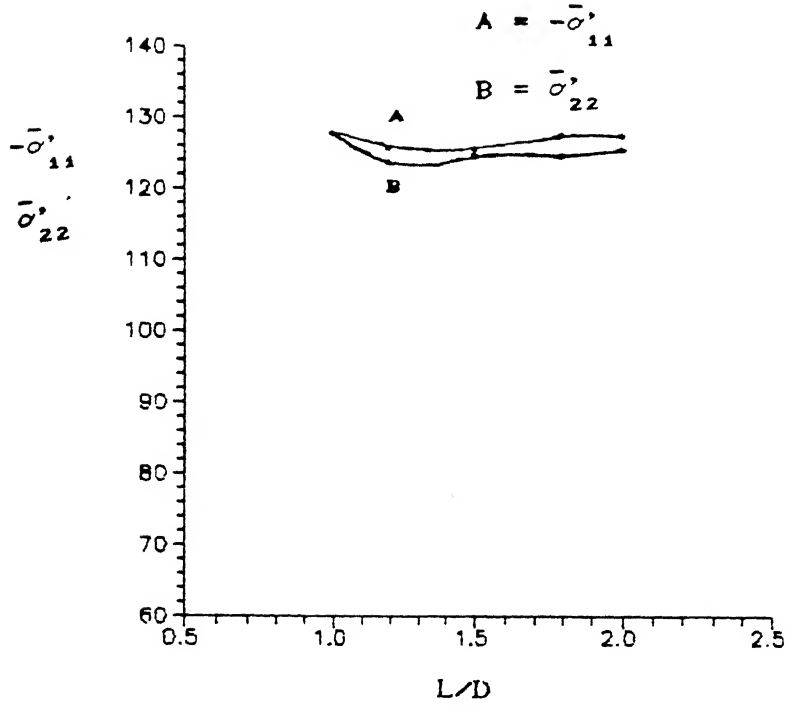


Fig.4.22 Variation of averaged stress components with L/D (V_f = 0.44, K_2/K_1 = 4.0, m = 0.5)

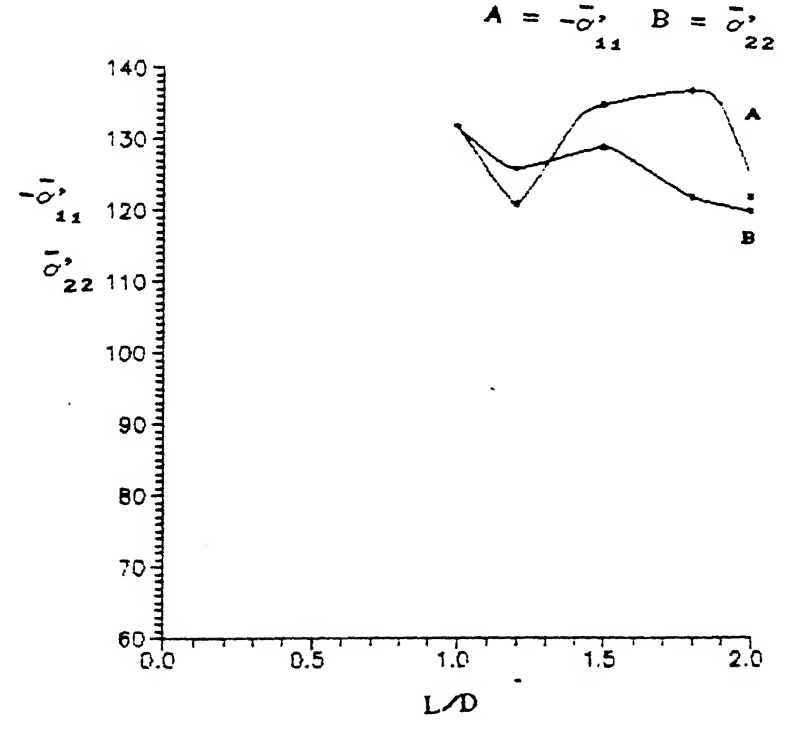


Fig.4.23 Variation of averaged stress components with L/D (V_f = 0.44, K_2/K_1 = 4.0, m = 0.0)

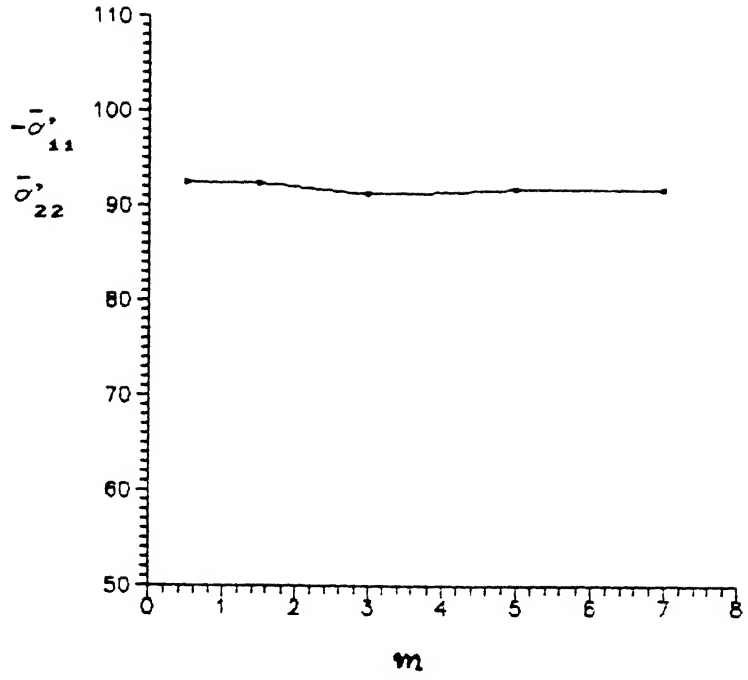


Fig.4.24 Variation of averaged stress components with m (V_f = 0.24, K_2/K_1 = 4.0, L/D = 1.5)

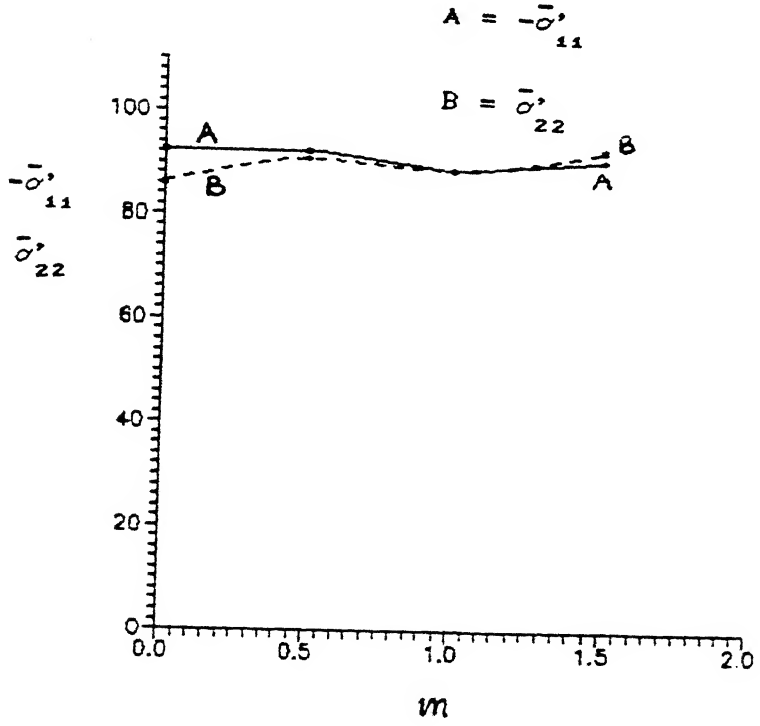


Fig.4.25 Variation of averaged stress components with m (V_f = 0.24, K_2/K_1 = 4.0, L/D = 2.0)

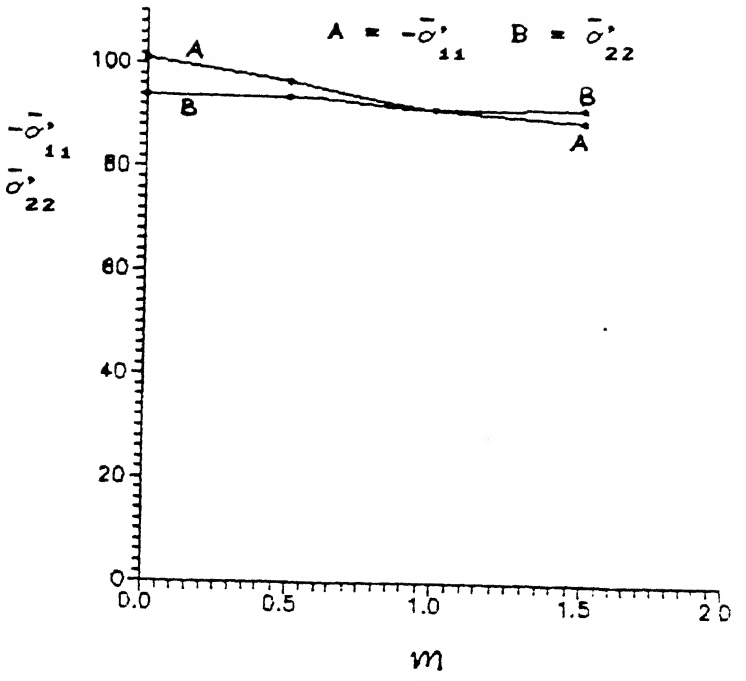


Fig.4.26 Variation of averaged stress components with m (V_f = 0.44, K_2/K_1 = 4.0, L/D = 2.0)

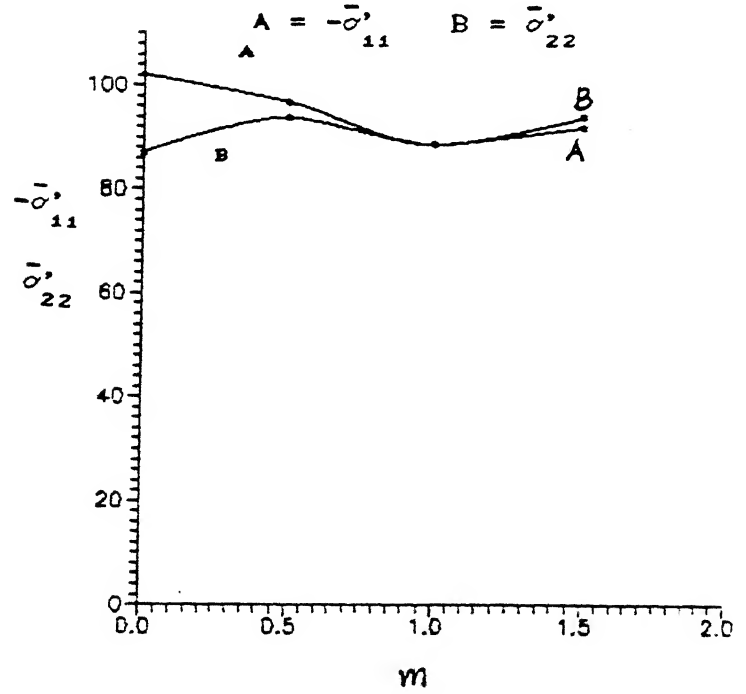


Fig.4.27 Variation of averaged stress components with m (V_f = 0.44, K_2/K_1 = 4.0, L/D = 3.0)

final microstructure

(b) assessment of load requirements.

(a) During transformation of initial microstructure to a final microstructure, two significant phenomenon which occur are shape change of second phase particle and dynamic recovery and recrystallization (changes taking place in matrix phase). As evident from Figs. 4.1 to 4.9 the deformation behaviour is extremely complex because of high degree of heterogeneity of stress and strain rate fields at microscopic level.

The instantaneous values of σ'_{22} (and consequently σ'_{11}) as shown in Figs. 4.2a and 4.2b vary in somewhat sinusoidal manner. Since the deviatoric stress component is responsible for shape change this would imply that the shape of second phase plate like particles will evolve into a non-uniform one.

The present model can be effectively used for a rough judgement of the range of heterogeneity in the stress and strain rate fields during the hot deformation of two-phase alloys with plate like second phase particle by setting the values of K_2/K_1 depending on material and trying out a range of V_f and L/D depending on the microstructure of the alloy. This is very important in connection with the phenomena of dynamic recovery and recrystallization occurring during the hot deformation of steel and Ti-alloys ($\alpha+\beta$). In fact temperature of deformation processing on which the present model parameters like K_2/K_1, V_f and L/D depend to a large extent can give significant informations using the present model. From the results of the present model, it can be seen that on the average strain rate in

the matrix phase is much higher than that in second phase plate and the range of heterogeneity of strain rate field in the second plates is much lower compared to that in the matrix phase. This is true for both K_2/K_1 as high as 4 (Figs. 4.5 and 4.6) and as low as 1.5 (Figs. 4.8 and 4.9) which is close to the value of K_2/K_1 in alloys like Ti-6Al-4V. This can be probable cause of the fact that during the thermomechanical processing of coarse two-phase alloys, one of the phase softens by both dynamic recovery and recrystallization whereas the other phase undergoes only recovery[8].

(b) It has been discussed in chapter 2 that the knowledge of material properties used in constitutive equation of the two phase material is an essential requirement to assess the load requirements. These material properties as stated in chapter 1 are sensitive to the microstructural parameters like second phase morphology which changes with the deformation, making the assessment of load requirement very difficult. So it is very important to correlate the material properties used in the constitutive equation (or in governing equation) to the microstructural parameters like plate size and shapes.

As stated in chapter 2 microstructure of the two-phase alloy as shown in Fig.(2.1a) is similar to the composite shown in Fig.(2.1b). Since the constitutive equation for a given two-phase structure (i.e. for fixed K_2/K_1 ratio, L/D ratio, volume fraction and plate orientation) must remain same for different flow fields, it must be expected that a given homogenization scheme must yield averaged stress values, as per computed by section

3.5, which are invariant to the flow field and boundary conditions. μ_{eff} (which is the material property for viscoplastic flow) calculated using $\bar{\sigma}'_{11}$ and $\bar{\sigma}'_{22}$ as discussed in section 4.3.1 has been found to be invariant of flow and the μ_{eff} thus calculated using the present model for different zones, can be used to calculate the μ_{eff} for the microstructure in Fig.(2.1b) by using the FEM program described in section 3.5 with minor modifications.

CHAPTER 5

CONCLUSIONS

High temperature plastic deformation behaviour of two-phase alloys having the second phase with plate morphology has been modelled using Finite Element Method (FEM). Constitutive equation for high temperature plastic deformation has been taken to be that of a viscoplastic material, i.e. the flow stress $\bar{\sigma}$ is related to strain-rate $\dot{\varepsilon}$ by the equation $\bar{\sigma} = K \bar{\dot{\varepsilon}}^{m'}$ and pseudo viscosity of the material μ is given as $\mu = K \bar{\dot{\varepsilon}}^{m'-1}$, where K and m are material constants.The variable parameters of the model were K_2/K_1, the aspect ratio, volume fraction and plate orientation of the second phase. Boundary condition has been also varied. The following conclusions can be drawn from the results of the present study :

(1) Plots of various stress and strain-rate components over the domain, show a high level of heterogeneity which exists in the vicinity of second phase plate like particles.

(2) Heterogeneity is more for the strain-rate components. On the average strain-rate is much higher in the softer matrix phase. This finding can perhaps explain the fact that during hot deformation of two-phase alloys dynamic recovery is observed in both the phases whereas dynamic recrystallization is restricted to only one phase.

(3) The nature of variation in the deviatoric stress component shows a somewhat sinusoidal nature at the two-phase boundary. This indicates that the shape change in the second

phase plate like particles will be non-uniform.

(4) Pseudo-viscosity, μ, in the viscoplastic deformation model, calculated using $\bar{\sigma}'_{22}$, $\bar{\sigma}'_{11}$, $\dot{\epsilon}_{11}$ and $\dot{\epsilon}_{22}$, can be treated as the effective pseudo-visocity, μ_{eff}, of the two-phase structure. This fact can be utilized for calculating μ_{eff} for a real microstructures in two-phase alloys. The assessment of load requirement for carrying out the plastic deformation of two-phase alloys can thus be made from the microstructural information.

(5) Results showing the variations in $\bar{\sigma}'_{11}$ and $\bar{\sigma}'_{22}$ with model parameter indicate the following,

(i) if μ_{eff} for a given morphology is known for two values of K_2/K_1 the μ_{eff} for any two-phase structure with same morphology and different K_2/K_1 can be conveniently interpolated

(ii) for $V_f < 0.25$, the effect of morphology of second phase on μ_{eff} is negligible.

REFERENCES

1. J. Gurland, Israel Journal of Technology, 24, (1988), p.243.
2. S. Ankem and H. Margolin, Met. Trans.A, 17A, (1986), p.2209.
3. J.C. Williams and E.A. Starke Jr."Deformation Processing and Structure", Ed. G. Krauss, ASM Metal Parks, Ohio, (1984), p.279.
4. J.E. Coyne, "Science, Technology and Application of Titanium", Ed. R.I. Jaffee and N.E. Promisel, Pergaman Press, Oxford, U.K., (1970), p.97.
5. S. Ankem and H. Margolin, Met. Trans.A, 13A, (1982), p.595.
6. S.M.L. Sastry, P.S. Rao and K.K. Sankaran, "Titanium Science and Technology", 4th. Int. Conf. Proc. on Titanium, Kyoto, Japan, 2, (1980), p.873.
7. I. Weiss, F.H. Froes, D. Eylon and G.E. Welsch, Met. Trans.A, 17A, (1986), p.1935.
8. W. Roberts, in ref.3, p.109.
9. George E. Dieter, "Mechanical Metallurgy", 3rd. edition, Mc.Graw-Hill Book Co.
10. K. Mori, K. Osakada and T. Oda, Int. J. Mech. Sci., 24, No.9, (1982), p.519.
11. T.L. Dragone and W.D. Nix, Acta. Metall. Mater., 38, No.10, (1990), p.1941.
12. Zaoui, "Plasticity Today : Modelling, Methods and Application", Ed. A. Sawczuk and G. Bianchi, Elsevier, London, 1985.
13. C. Taylor and T.G. Hughes, "Finite Element Programming of Navier-Stokes Equations", Pineridge Press Ltd., Swansea,U.K.

Appendix A

```
a 0.5 vf=.44 l/d=1.5 4.0
0 34 8 8 1 1 90.0
0 86 0 0
1 .5 .01 100 0 0
0.5 .4062 .2708 4.0
209 3 4
209 1 4
210 3 4
210 1 4
211 3 4
211 1 4
212 3 4
212 1 4
213 3 4
213 1 4
214 3 4
214 1 4
215 3 4
215 1 4
216 3 4
216 1 4
217 3 4
217 1 4
218 3 4
218 1 4
219 3 4
219 1 4
220 3 4
220 1 4
221 3 4
221 1 4
222 3 4
222 1 4
223 3 4
```

```
223 1 4
224 3 4
224 1 4
225 3 4
225 1 4
1 3 0
2 3 0
3 3 0
4 3 0
5 3 0
6 3 0
7 3 0
8 3 0
9 3 0
10 3 0
11 3 0
12 3 0
13 3 0
14 3 0
15 3 0
16 3 0
17 3 0
1 1 2
18 1 2
27 1 2
44 1 2
53 1 2
70 1 2
79 1 2
96 1 2
```

```
105 1 2
122 1 2
131 1 2
148 1 2
157 1 2
174 1 2
183 1 2
200 1 2
209 1 2
17 1 -2
26 1 -2
43 1 -2
52 1 -2
69 1 -2
78 1 -2
95 1 -2
104 1 -2
121 1 -2
130 1 -2
147 1 -2
156 1 -2
173 1 -2
182 1 -2
199 1 -2
208 1 -2
225 1 -2
1 2 0.0
```

```
a 0.5 vf=.44 l/d=1.5 4.0
0 39 8 8 1 1 90.0
0 102 0 0
1 .5 .01 1 100 0 0
0.5 .4062 .2708 4.0
209 3 4
209 1 2
210 3 4
210 1 2
211 3 4
211 1 2
212 3 4
212 1 2
213 3 4
213 1 2
214 3 4
214 1 2
215 3 4
215 1 2
216 3 4
216 1 2
217 3 4
217 1 2
218 3 4
218 1 2
219 3 4
219 1 2
220 3 4
220 1 2
221 3 4
221 1 2
222 3 4
222 1 2
223 3 4
223 1 2
224 3 4
224 1 2
225 3 4
225 1 2
1 3 0
2 3 0
3 3 0
4 3 0
5 3 0
6 3 0
7 3 0
8 3 0
9 3 0
10 3 0
11 3 0
12 3 0
13 3 0
14 3 0
15 3 0
16 3 0
17 3 0
1 1 2
18 1 2
27 1 2
44 1 2
53 1 2
70 1 2
79 1 2
96 1 2
105 1 2
122 1 2
```

```
131 1 2
148 1 2
157 1 2
174 1 2
183 1 2
200 1 2
209 1 2
17 1 -2
17 3  2
26 1 -2
26 3 2
43 1 -2
43 3 2
52 1 -2
52 3 2
69 1 -2
69 3 2
78 1 -2
78 3 2
95 1 -2
95 3 2
104 1 -2
104 3 2
121 1 -2
121 3 2
130 1 -2
130 3 2
147 1 -2
147 3 2
156 1 -2
156 3 2
173 1 -2
173 3 2
182 1 -2
182 3 2
199 1 -2
199 3 2
208 1 -2
208 3 2
225 1 -2
1 2 0.0
```

```
a  0.5  vf=.44  l/d=1.5  4.0
0  34  8  8  1  1  90.0
0  86  0  0
1  .5  .01  100  0  0
0.5  .4062  .2708  4.0
209  3  4
209  1  5
210  3  4
210  1  5
211  3  4
211  1  5
212  3  4
212  1  5
213  3  4
213  1  5
214  3  4
214  1  5
215  3  4
215  1  5
216  3  4
216  1  5
217  3  4
217  1  5
218  3  4
218  1  5
219  3  4
219  1  5
220  3  4
220  1  5
221  3  4
221  1  5
222  3  4
222  1  5
223  3  4
223  1  5
224  3  4
224  1  5
225  3  4
225  1  5
1  3  0
2  3  0
3  3  0
4  3  0
5  3  0
6  3  0
7  3  0
8  3  0
9  3  0
10  3  0
11  3  0
12  3  0
13  3  0
14  3  0
15  3  0
16  3  0
17  3  0
1  1  2
18  1  2
27  1  2
44  1  2
53  1  2
70  1  2
79  1  2
96  1  2
105  1  2
```

```
122 1 2
131 1 2
148 1 2
157 1 2
174 1 2
183 1 2
200 1 2
209 1 2
17 1 -2
26 1 -2
43 1 -2
52 1 -2
69 1 -2
78 1 -2
95 1 -2
104 1 -2
121 1 -2
130 1 -2
147 1 -2
156 1 -2
173 1 -2
182 1 -2
199 1 -2
208 1 -2
225 1 -2
1 2 0.0
```

Appendix B

```
      implicit real*8(a-h,o-z)
      DIMENSION BOUDV(624),COORD(264,2),EQRHS(624),GFLUM(105,105),
     1          GRADB(264,4),LBOUD(624),LHEDV(105),LNODS(73,8),
     1          NADFM(264),NGRAD(73),NODFM(264),PNORM(105),
     1          POSGP(3),VARB1(624),VARB2(624),WEIGP(3)
     1          ,STR11(73),STR22(73),STR12(73),VALT(73,9)
      OPEN(UNIT=7,FILE='out8')
      OPEN(UNIT=40,FILE='i80.inp')
      OPEN(UNIT=10,FILE='val15.dat',form='unformatted')
      OPEN(UNIT=25,FILE='val16.dat',form='unformatted')
      open(unit=8,file='sig')
      open(unit=9,file='sref')
      open(unit=11,file='tao')
      open(unit=12,file='tao11')
      open(unit=13,file='tao22')
      open(unit=14,file='pres')
      open(unit=15,file='sr11')
      open(unit=16,file='sr22')
      open(unit=17,file='sr12')
      open(unit=18,file='tao12')

C     SET DYNAMIC DIMENSION VALUES

      CALL DIMENS(MELEM,MFRON,MPOIN,MTOTV)

C     READ IN ALL PROBLEM DATA


      CALL DINPUT(BOUDV,COORD,DENSY,GRADB,IAXSY,LBOUD,LNODS,
     1            MELEM,MPOIN,MTOTV,NADFM,NBCON,NDOFM,NELEM,
     1            NEVAB,NGAUS,NGRAD,NITER,NNODL,NNODP,NODFM,
     1            NPOIN,NTOTV,RELAX,TOLER,VARB1,VARB2,VISCY,
     1            XFORC,YFORC,SLOPE1,D,E,FAC,XMAX,YMAX)
C


C     CALCULATE THE SHAPE FUNCTION AND DERIVATIVE VALUES
C
      CALL DRIVES(COORD,LNODS,MELEM,MPOIN,NELEM,NGAUS,NNODL,
     1            NNODP,POSGP,WEIGP)
C
      CALL MATPRT(COORD,VALT,LNODS,MELEM,MPOIN,NELEM,NGAUS,
     1 POSGP,XMAX,YMAX,SLOPE1,D,E,FAC,NNODP)
C     SET UP EQUATIONS AND ITERATE UNTIL SOLUTIONS CONVERGE
C
      CALL ITERAT(BOUDV,COORD,DENSY,EQRHS,GFLUM,GRADB,IAXSY,
     1            LBOUD,LHEDV,LNODS,MELEM,MFRON,MPOIN,MTOTV,
     1            NADFM,NBCON,NDOFM,NELEM,NEVAB,NGAUS,NGRAD,
     1            NITER,NNODL,NNODP,NODFM,NPOIN,NTOTV,PNORM,
     1            POSGP,RELAX,TOLER,VARB1,VARB2,VISCY,WEIGP,
     1            XFORC,YFORC,VALT)
      write(*,*) 'no'
      CALL SRCAL(VARB1,LNODS,NADFM,NODFM,MTOTV,MPOIN,MELEM,
     1 NPOIN,NELEM,NNODP,NNODL,STR11,STR22,STR12,VISCY,VALT
     1 ,COORD,posgp)
```

```
      write(*,*) 'yes'
      CLOSE(UNIT=40)
      CLOSE(UNIT=10)
      CLOSE(UNIT=25)
      CLOSE(UNIT=7)
      close(unit=8)
      close(unit=9)
      close(unit=11)
      close(unit=12)
      close(unit=13)
      close(unit=14)
      close(unit=15)
      close(unit=16)
      close(unit=17)
      close(unit=18)
      STOP
      END

      SUBROUTINE DIMENS(MELEM,MFRON,MPOIN,MTOTV)
      MELEM=73
      MFRON=105
      MPOIN=264
      MTOTV=624
      RETURN
      END

      SUBROUTINE SHAPE4(DERIV,SHAPE,XEQIV,YEQIV)
      DIMENSION DERIV(2,8),SHAPE(8)
C
C     LINEAR ELEMENT ANTI-CLOCKWISE NODE NUMBERING
C
      X=XEQIV
      Y=YEQIV
      XY=X*Y
      SHAPE(1)=(1-X-Y+XY)*0.25
      SHAPE(2)=(1+X-Y-XY)*0.25
      SHAPE(3)=(1+X+Y+XY)*0.25
      SHAPE(4)=(1-X+Y-XY)*0.25
      DERIV(1,1)=(-1+Y)*0.25
      DERIV(1,2)=(1-Y)*0.25
      DERIV(1,3)=(1+Y)*0.25
      DERIV(1,4)=(-1-Y)*0.25
      DERIV(2,1)=(-1+X)*0.25
      DERIV(2,2)=(-1-X)*0.25
      DERIV(2,3)=(1+X)*0.25
      DERIV(2,4)=(1-X)*0.25
      RETURN
      END
C
C
C
      SUBROUTINE SHAPE8(DERIV,SHAPE,XEQIV,YEQIV)
      DIMENSION DERIV(2,8),SHAPE(8)
C
C     PARABOLIC ELEMENT ANTI-CLOCKWISE NODE NUMBERING
```

```
C
      X=XEQIV
      Y=YEQIV
      XY=X*Y
      XX=X*X
      YY=Y*Y
      XXY=XX*Y
      XYY=X*YY
      X2=X*2
      Y2=Y*2
      XY2=XY*2
      SHAPE(1)=(-1+XY+XX+YY-XXY-XYY)*0.25
      SHAPE(2)=(1-Y-XX+XXY)*0.5
      SHAPE(3)=(-1-XY+XX+YY-XXY+XYY)*0.25
      SHAPE(4)=(1+X-YY-XYY)*0.5
      SHAPE(5)=(-1+XY+XX+YY+XXY+XYY)*0.25
      SHAPE(6)=(1+Y-XX-XXY)*0.5
      SHAPE(7)=(-1-XY+XX+YY+XXY-XYY)*0.25
      SHAPE(8)=(1-X-YY+XYY)*0.5
      DERIV(1,1)=(Y+X2-XY2-YY)*0.25
      DERIV(1,2)=(-X+XY)
      DERIV(1,3)=(-Y+X2-XY2+YY)*0.25
      DERIV(1,4)=(1-YY)*0.5
      DERIV(1,5)=(Y+X2+XY2+YY)*0.25
      DERIV(1,6)=-X-XY
      DERIV(1,7)=(-Y+X2+XY2-YY)*0.25
      DERIV(1,8)=(-1+YY)*0.5
      DERIV(2,1)=(X+Y2-XX-XY2)*0.25
      DERIV(2,2)=(-1+XX)*0.5
      DERIV(2,3)=(-X+Y2-XX+XY2)*0.25
      DERIV(2,4)=-Y-XY
      DERIV(2,5)=(X+Y2+XX+XY2)*0.25
      DERIV(2,6)=(1-XX)*0.5
      DERIV(2,7)=(-X+Y2+XX-XY2)*0.25
      DERIV(2,8)=-Y+XY
      RETURN
      END
C
C
C
      SUBROUTINE DJACOB(COORD,DERIV,DETJB,DJACI,DJACK,IELEM,LNODS,
     1            MELEM,MPOIN,NNODP)
      DIMENSION COORD(MPOIN,2) ,DERIV(2,8),DJACI(2,2),DJACK(2,2),
     1   LNODS(MELEM,8)
C
C     SET UP TEMPORARY MATRIX TO ALLOW JACOBIAN TO BE FORMED
C
      DO 20 IDIME=1,2
      DO 20 JDIME=1,2
      TEMPY=0
      DO 10 INODP=1,NNODP
      KPOIN=IABS(LNODS(IELEM,INODP))
      TEMPY=TEMPY+DERIV(IDIME,INODP)*COORD(KPOIN,JDIME)
10    CONTINUE
      DJACK(IDIME,JDIME)=TEMPY
```

```
20      CONTINUE
        DETJB=DJACK(1,1)*DJACK(2,2)-DJACK(2,1)*DJACK(1,2)
C
C       CHECK FOR ZERO OR NEGATIVE DETERMINANT
C
        IF(DETJB)30,30,40
30      CONTINUE
        WRITE(7,2000)IELEM
        STOP
40      CONTINUE
C
C       INVERT TEMPORARY MATRIX TO FORM JACOBIAN-INVERSE
C
        DJACI(1,1)=DJACK(2,2)/DETJB
        DJACI(2,2)=DJACK(1,1)/DETJB
        DJACI(1,2)=-DJACK(1,2)/DETJB
        DJACI(2,1)=-DJACK(2,1)/DETJB
2000    FORMAT(//37H NON POSITIVE DETERMINANT FOR ELEMENT,I4)
        RETURN
        END
C
C
C
        SUBROUTINE DRIVES(COORD,LNODS,MELEM,MPOIN,NELEM,NGAUS,NNODL,
     1            NNODP,POSGP,WEIGP)
C
C       EXTERNAL SUBROUTINES:
C       SHAPE8,DJACOB,SHAPE4
C
        DIMENSION AREAW(9),CARLG(2,36),CARPG(2,72),CARTP(2,8),
     1   DERIV(2,8),DJACI(2,2),DJACK(2,2),POSGP(3),COORD(MPOIN,2),
     2   SHALG(36),SHAPE(8),SHAPG(72),WEIGP(3),LNODS(MELEM,8)
C
C       REWIND TAPE BEFORE WRITING ON SHAPE FUNCTIONS
C
        REWIND 10
C
C       SET UP POSITIONS AND WEIGHTS FOR 3 POINT GAUSS RULE
C
        POSGP(1)=0.7745966692
        POSGP(2)=0
        POSGP(3)=-POSGP(1)
        WEIGP(1)=0.5555555556
        WEIGP(2)=0.8888888889
        WEIGP(3)=WEIGP(1)
C
C       CALCULATE SHAPE FUNCTIONS AND LOCAL DERIVATIVES FOR ELEMENTS
C
        DO 60 IELEM=1,NELEM
        LGAUS=0
        DO 50 IGAUS=1,NGAUS
        DO 50 JGAUS=1,NGAUS
        LGAUS=LGAUS+1
        XEQIV=POSGP(IGAUS)
        YEQIV=POSGP(JGAUS)
```

```
C
C       USE GAUSS POSITIONS TO CALCULATE LOCAL VALUES
C
        CALL SHAPE8(DERIV,SHAPE,XEQIV,YEQIV)
C
C       SET UP JACOBIAN MATRIX AND INVERSE
C
        IELEMS=IELEM
        CALL DJACOB(COORD,DERIV,DETJB,DJACI,DJACK,IELEMS,LNODS,
     1     MELEM,MPOIN,NNODP)
C
C       CALCULATE GLOBAL DERIVATIVES AND AREA*GAUSS WEIGHTS
C
        DO 10 IDIME=1,2
        DO 10 INODP=1,NNODP
        CARTP(IDIME,INODP)=0
        DO 10 JDIME=1,2
        CARTP(IDIME,INODP)=CARTP(IDIME,INODP)+
     1  DJACI(IDIME,JDIME)*DERIV(JDIME,INODP)
10      CONTINUE
        AREAW(LGAUS)=DETJB*WEIGP(IGAUS)*WEIGP(JGAUS)
C
C       PUT SHAPE FUNCTIONS AND DERIVATIVES IN ELEMENT MATRIX
C
        DO 20 INODP=1,NNODP
        KGAPA=(LGAUS-1)*NNODP+INODP
        SHAPG(KGAPA)=SHAPE(INODP)
        DO 20 IDIME=1,2
        CARPG(IDIME,KGAPA)=CARTP(IDIME,INODP)
20      CONTINUE
C
C       USE GAUSS POSITIONS TO CALCULATE LOCAL VALUES
C
        CALL SHAPE4(DERIV,SHAPE,XEQIV,YEQIV)
C
C       CALCULATE GLOBAL DERIVATIVES FOR LINEAR FUNCTIONS
C
        DO 30 IDIME=1,2
        DO 30 INODL=1,NNODL
        CARTP(IDIME,INODL)=0
        DO 30 JDIME=1,2
        CARTP(IDIME,INODL)=CARTP(IDIME,INODL)+
     1  DJACI(IDIME,JDIME)*DERIV(JDIME,INODL)
30      CONTINUE
C
C       PUT SHAPE FUNCTIONS AND DERIVATIVES IN ELEMENT MATRIX
C
        DO 40 INODL=1,NNODL
        KGALI=(LGAUS-1)*NNODL+INODL
        SHALG(KGALI)=SHAPE(INODL)
        DO 40 IDIME=1,2
        CARLG(IDIME,KGALI)=CARTP(IDIME,INODL)
40      CONTINUE
50      CONTINUE
C
```

```
C       NOW WRITE ALL THE ELEMENT SHAPE FUNCTIONS ON TAPE
C
        NGAPA=LGAUS*NNODP
        NGALI=LGAUS*NNODL
        WRITE(10) IELEM,LGAUS,NGAPA,NGALI,
     1((CARPG(IDIME,IGAPA),IDIME=1,2),SHAPG(IGAPA),IGAPA=1,NGAPA),
     2((CARLG(IDIME,IGALI),IDIME=1,2),SHALG(IGALI),IGALI=1,NGALI),
     3(AREAW(IGAUS),IGAUS=1,LGAUS)
60      CONTINUE
        RETURN
        END
C
C
C
C
C
        SUBROUTINE SRCAL(VARB1,LNODS,NADFM,NODFM,MTOTV,MPOIN,MELEM,
     1  NPOIN,NELEM,NNODP,NNODL,STR11,STR22,STR12,VISCY,VALT,COORD
     1  ,posgp)

        DIMENSION STR11(MELEM),STR22(MELEM),STR12(MELEM)
     1  ,CARTP(2,8),NODFM(MPOIN),NADFM(MPOIN),LNODS(MELEM,8)
     1  ,VARB1(MTOTV),SHAPG(72),CARPG(2,72),UVEL(8),VVEL(8),
     1  SHAPL(4),SHAPP(8),SHALG(36),CARTL(2,4),CARLG(2,36)
     1  ,AREAW(9),VALT(MELEM,9),COORD(MPOIN,2),sr11(melem,9),
     1  sr22(melem,9),sr12(melem,9),t11(melem,9),teq(melem,9),
     1  t22(melem,9),t12(melem,9),press(melem,9),sref(melem,9),
     1  sig11(melem,9),sig22(melem,9),sig12(melem,9),sigeq(melem,9)
     1  ,deriv(2,8),shapx(8),posgp(3),gcoord1(melem,9),
     1   gcoord2(melem,9),areawe(melem,9)

        REWIND(10)
        DO 110 IELEM=1,NELEM
        STR11(IELEM)=0
        STR22(IELEM)=0
        STR12(IELEM)=0
        READ(10) JELEM,LGAUS,NGAPA,NGALI,
     1((CARPG(IDIME,IGAPA),IDIME=1,2),SHAPG(IGAPA),IGAPA=1,NGAPA),
     2((CARLG(IDIME,IGALI),IDIME=1,2),SHALG(IGALI),IGALI=1,NGALI),
     3(AREAW(IGAUS),IGAUS=1,LGAUS)
        write(*,*) 'lgaus=',lgaus
        DO 100 IGAUS=1,LGAUS
        DAREA=AREAW(IGAUS)
        areawe(ielem,igaus)=darea
        DO 30 INODP=1,NNODP
        SHAPP(INODP)=SHAPG(NNODP*(IGAUS-1)+INODP)
        DO 30 IDIME=1,2
        CARTP(IDIME,INODP)=CARPG(IDIME,NNODP*(IGAUS-1)+INODP)
30      CONTINUE
        DO 40 INODL=1,NNODL
        SHAPL(INODL)=SHALG(NNODL*(IGAUS-1)+INODL)
        DO 40 IDIME=1,2
        CARTL(IDIME,INODL)=CARLG(IDIME,NNODL*(IGAUS-1)+INODL)
40      CONTINUE
        UVELY=0
```

```
      VVELY=0
      srll(ielem,igaus)=0
      sr22(ielem,igaus)=0
      sr12(ielem,igaus)=0
      sref(ielem,igaus)=0
C
      do 41 inodp=1,nnodp
      KPOIN=IABS(LNODS(IELEM,INODP))
      ITOTU=NADFM(KPOIN)
      ITOTV=ITOTU+NODFM(KPOIN)-1
      CARX=CARTP(1,INODP)
      CARY=CARTP(2,INODP)
      srll(ielem,igaus)=varbl(itotu)*carx+srll(ielem,igaus)
      sr22(ielem,igaus)=varbl(itotv)*cary+sr22(ielem,igaus)
      sr12(ielem,igaus)=varbl(itotu)*cary*.5+varbl(itotv)*carx*.5
     1 +sr12(ielem,igaus)
41    continue
      srllav=srllav+srll(ielem,igaus)*darea
      sr22av=sr22av+sr22(ielem,igaus)*darea
      sr12av=sr12av+sr12(ielem,igaus)*darea
      sref(ielem,igaus)=sqrt(2*(srll(ielem,igaus)**2)
     1 +(sr12(ielem,igaus)**2)
     1 +2*(sr22(ielem,igaus)**2))
      srefav=srefav+sref(ielem,igaus)*darea
      vsfac=sref(ielem,igaus)**-.95
      tll(ielem,igaus)=vsfac*valt(ielem,igaus)*viscy*
     1 srll(ielem,igaus)
      t22(ielem,igaus)=vsfac*valt(ielem,igaus)*viscy*
     1 sr22(ielem,igaus)
      tl2(ielem,igaus)=vsfac*valt(ielem,igaus)*viscy*
     1 sr12(ielem,igaus)
      teq(ielem,igaus)=valt(ielem,igaus)*viscy*vsfac*
     1 sref(ielem,igaus)
      teqav=teqav+teq(ielem,igaus)*darea
C     EVALUATE  VELOCITIES & STRAIN-RATES AT GAUSS POINTS


c     EVALUATE HYDROSTATIC STRESS AT GAUSS-POINTS
      press(ielem,igaus)=0.0
      DO 49 INODL=1,NNODL
      JNODL=2*INODL-1
      KPOIN=IABS(LNODS(IELEM,JNODL))
      ITOTU=NADFM(KPOIN)
      ITOTP=ITOTU+1
      press(ielem,igaus)=press(ielem,igaus)
     1 +varbl(itotp)*shapl(inodl)
49    CONTINUE
      pressav=pressav+press(ielem,igaus)*darea
C
      DO 50 INODP=1,NNODP
      KPOIN=IABS(LNODS(IELEM,INODP))
      ITOTU=NADFM(KPOIN)
      ITOTV=ITOTU+NODFM(KPOIN)-1
      SHAPE=SHAPP(INODP)
      CARX=CARTP(1,INODP)
```

```
      CARY=CARTP(2,INODP)
      UVELY=UVELY+VARB1(ITOTU)*SHAPE
      VVELY=VVELY+VARB1(ITOTV)*SHAPE
      STR11(IELEM)=STR11(IELEM)+VARB1(ITOTU)*CARX*DAREA*VISCY
     1*VALT(IELEM,IGAUS)*vsfac
      STR22(IELEM)=STR22(IELEM)+VARB1(ITOTV)*CARY*DAREA*VISCY
     1*VALT(IELEM,IGAUS)*vsfac
      STR12(IELEM)=STR12(IELEM)+0.5*(VARB1(ITOTU)*CARY
     1 +VARB1(ITOTV)*CARX)*DAREA*VISCY*VALT(IELEM,IGAUS)*vsfac
50    continue
100   continue
110   continue
      do 112 ielem=1,nelem
      do 111 kgaus=1,9
      sig11(ielem,kgaus)=-press(ielem,kgaus)+t11(ielem,kgaus)
      sig22(ielem,kgaus)=-press(ielem,kgaus)+t22(ielem,kgaus)
      sig12(ielem,kgaus)=t12(ielem,kgaus)
      sigeq(ielem,kgaus)=sqrt((sig11(ielem,kgaus)-sig22(ielem
     1,kgaus))**2.0+6.0*(sig12(ielem,kgaus)**2.0)
     1 +sig11(ielem,igaus)**2.0 )
      sigeqav=sigeqav+sigeq(ielem,igaus)*areawe(ielem,igaus)
111   continue
112   continue
      do 115 ielem=1,nelem
      lgaus=0
      do 114 igaus=1,3
      do 113 jgaus=1,3
      xeqiv=posgp(igaus)
      yeqiv=posgp(jgaus)
      lgaus=lgaus+1
      call shape8(deriv,shapx,xeqiv,yeqiv)
      gcoord1(ielem,lgaus)=0.0
      gcoord2(ielem,lgaus)=0.0
      do 1130 inodp=1,nnodp
      kpoin=iabs(lnods(ielem,inodp))
      write(70,*) kpoin
      gcoord1(ielem,lgaus)=gcoord1(ielem,lgaus)+coord(kpoin,1)
     1*shapx(inodp)
      gcoord2(ielem,lgaus)=gcoord2(ielem,lgaus)+coord(kpoin,2)
     1*shapx(inodp)
      write(70,*) shapx(inodp),coord(kpoin,1),coord(kpoin,2)
1130  continue
113   continue
114   continue
115   continue
      av11=0
      av22=0
      av12=0
      DO 120 IELEM=1,NELEM
      WRITE(7,*) IELEM,STR11(IELEM),STR22(IELEM),STR12(IELEM)
      av11=av11+str11(ielem)
      av22=av22+str22(Ielem)
      av12=av12+str12(ielem)
120   CONTINUE
      WRITE(7,*) sr11av,sr22av,sr12av,srefav,teqav,av11,av22,av12
```

```
      WRITE(*,*) sr11av,sr22av,sr12av,srefav,teqav,av11,av22,av12
      do 125 ielem=1,nelem
      do 124 igaus=1,9
c       write(8,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1 ,sigeq(ielem,igaus)
c       write(9,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1 ,sref(ielem,igaus)
c       write(11,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1 ,teq(ielem,igaus)
c       write(12,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1          ,t11(ielem,igaus)
c       write(13,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1 ,t22(ielem,igaus)
c       write(14,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1 ,press(ielem,igaus)
c       write(15,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1 ,sr11(ielem,igaus)
c       write(16,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1 ,sr22(ielem,igaus)
c       write(17,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1 ,sr12(ielem,igaus)
c       write(18,*) gcoord1(ielem,igaus),gcoord2(ielem,igaus)
c     1 ,t12(ielem,igaus)
124     continue
125     continue
c       write(8,*) sigeqav
        write(9,*) srefav
        write(11,*) teqav
        write(15,*) sr11av
        write(16,*) sr22av
        RETURN
        END

C
C
C
        SUBROUTINE SURFIN(COORD,COSLX,COSLY,IELEM,IGAUS,ISIDE,LNODS,
     1            MELEM,MPOIN,NNODP,POSGP,SHAPE,SLETH,WEIGP)
C
C       EXTERNAL SUBROUTINES:
C       SHAPE8,DJACOB
C
        DIMENSION DERIV(2,8),DJACI(2,2),DJACK(2,2),POSGP(3),WEIGP(3),
     1   COORD(MPOIN,2),LNODS(MELEM,8),SHAPE(8)
        GO TO (10,20,30,40),ISIDE
C
C       SET UP PARAMETERS FOR ELEMENT SIDE WITH FIXED GRADIENTS
C
10      CONTINUE
        XEQIV=-1
        YEQIV=POSGP(IGAUS)
        ITEMP=2
        RTEMP=-1
        GO TO 50
20      CONTINUE
```

```
      XEQIV=POSGP(IGAUS)
      YEQIV=1
      ITEMP=1
      RTEMP=-1
      GO TO 50
30    CONTINUE
      XEQIV=1
      YEQIV=POSGP(IGAUS)
      ITEMP=2
      RTEMP=1
      GO TO 50
40    CONTINUE
      XEQIV=POSGP(IGAUS)
      YEQIV=-1
      ITEMP=1
      RTEMP=1
50    CONTINUE
C
C     DETERMINE SHAPE FUNCTION VALUES AND LOCAL DERIVATIVES
C
      CALL SHAPE8(DERIV,SHAPE,XEQIV,YEQIV)
C
C     EVALUATE JACOBIAN AND COSINES OF OUTWARD NORMALS
C
      IELEMS=IELEM
      CALL DJACOB(COORD,DERIV,DETJB,DJACI,DJACK,IELEMS,LNODS,
     1    MELEM,MPOIN,NNODP)
      TEMPY=SQRT(DJACK(ITEMP,1)**2+DJACK(ITEMP,2)**2)
      COSLX=RTEMP*DJACK(ITEMP,2)/TEMPY
      COSLY=-RTEMP*DJACK(ITEMP,1)/TEMPY
      SLETH=TEMPY*WEIGP(IGAUS)
      RETURN
      END
C
C
      SUBROUTINE PRESCR(COORD,EQRHS,GRADB,IAXSY,IELEM,LNODS,MELEM,
     1           MPOIN,MTOTV,NADFM,NGAUS,NGRAD,NNODP,NODFM,
     2           POSGP,VISCY,WEIGP)
C
C     EXTERNAL SUBROUTINES:
C     SURFIN
C
      DIMENSION COORD(MPOIN,2),EQRHS(MTOTV),GRADB(MPOIN,4),
     1  LNODS(MELEM,8),NADFM(MPOIN),NGRAD(MELEM),
     2  NODFM(MPOIN),SHAPE(8),POSGP(3),WEIGP(3)
      ISIDE=NGRAD(IELEM)
C
C     LOOP TO CARRY OUT GAUSS INTEGRATION
C
      DO 30 IGAUS=1,NGAUS
      IELEMS=IELEM
      IGAUSS=IGAUS
      CALL SURFIN(COORD,COSLX,COSLY,IELEMS,IGAUSS,ISIDE,LNODS,
     1    MELEM,MPOIN,NNODP,POSGP,SHAPE,SLETH,WEIGP)
      TDUDN=0
```

```
      RADUS=0
      TDVDN=0
C
C     EVALUATE (VISCY*NORMAL-VELOCITY-GRADIENTS) AT THE GAUSS POINT
C
      DO 10 INODP=1,NNODP
      KPOIN=IABS(LNODS(IELEM,INODP))
      TDUDN=TDUDN+VISCY*SHAPE(INODP)*
     1    (GRADB(KPOIN,1)*COSLX+GRADB(KPOIN,2)*COSLY)
      RADUS=RADUS+SHAPE(INODP)*COORD(KPOIN,2)
      TDVDN=TDVDN+VISCY*SHAPE(INODP)*
     1    (GRADB(KPOIN,3)*COSLX+GRADB(KPOIN,4)*COSLY)
10    CONTINUE
      IF(IAXSY.EQ.1)SLETH=SLETH*RADUS
C
C     INCLUDE BOUNDARY GRADIENT TERMS INTO GLOBAL RHS VECTOR
C
      DO 20 INODP=1,NNODP
      KPOIN=IABS(LNODS(IELEM,INODP))
      ITOTU=NADFM(KPOIN)
      ITOTV=ITOTU+NODFM(KPOIN)-1
      EQRHS(ITOTU)=EQRHS(ITOTU)+SHAPE(INODP)*TDUDN*SLETH
      EQRHS(ITOTV)=EQRHS(ITOTV)+SHAPE(INODP)*TDVDN*SLETH
20    CONTINUE
30    CONTINUE
      RETURN
      END
C
      SUBROUTINE MATRIX(COORD,DENSY,EQRHS,FLUMX,IAXSY,LNODS,MELEM,
     1          MPOIN,MTOTV,NADFM,NEVAB,NNODL,NNODP,NODFM,
     2          VARB2,VISCY,XFORC,YFORC,VALT,IITER)
      DIMENSION AREAW(9),CARLG(2,36),CARPG(2,72),CARTL(2,4),
     1  CARTP(2,8),COORD(MPOIN,2),EQRHS(MTOTV),ERHSU(8),
     2  ERHSV(8),FLUMX(20,20),LNODS(MELEM,8),NADFM(MPOIN),
     3  NODFM(MPOIN),SHALG(36),SHAPG(72),SHAPL(4),
     4  SHAPP(8),VARB2(MTOTV),VALT(MELEM,9)
C
C     INITIALIZE ARRAYS
C
      DO 10 INODP=1,NNODP
      ERHSU(INODP)=0
      ERHSV(INODP)=0
10    CONTINUE
      DO 20 IEVAB=1,NEVAB
      DO 20 JEVAB=1,NEVAB
      FLUMX(IEVAB,JEVAB)=0
20    CONTINUE
C
C     READ IN SHAPE FUNCTION VALUES AND DERIVATIVES
C
      READ(10) IELEM,LGAUS,NGAPA,NGALI,
     1((CARPG(IDIME,IGAPA),IDIME=1,2),SHAPG(IGAPA),IGAPA=1,NGAPA),
     2((CARLG(IDIME,IGALI),IDIME=1,2),SHALG(IGALI),IGALI=1,NGALI),
     3(AREAW(IGAUS),IGAUS=1,LGAUS)
C
```

```
C       LOOP TO CARRY OUT GAUSS INTEGRATION
C
        DO 100 IGAUS=1,LGAUS
        DAREA=AREAW(IGAUS)
        DO 30 INODP=1,NNODP
        SHAPP(INODP)=SHAPG(NNODP*(IGAUS-1)+INODP)
        DO 30 IDIME=1,2
        CARTP(IDIME,INODP)=CARPG(IDIME,NNODP*(IGAUS-1)+INODP)
30      CONTINUE
        DO 40 INODL=1,NNODL
        SHAPL(INODL)=SHALG(NNODL*(IGAUS-1)+INODL)
        DO 40 IDIME=1,2
        CARTL(IDIME,INODL)=CARLG(IDIME,NNODL*(IGAUS-1)+INODL)
40      CONTINUE
        UVELY=0
        RADUS=0
        VVELY=0
        SR11=0
        SR22=0
        SR12=0
        SREF=0
        VSFAC=1.0
C
C       EVALUATE RADIUS AND PREVIOUS VELOCITIES AT GAUSS POINTS
C
        DO 50 INODP=1,NNODP
        KPOIN=IABS(LNODS(IELEM,INODP))
        ITOTU=NADFM(KPOIN)
        ITOTV=ITOTU+NODFM(KPOIN)-1
        SHAPE=SHAPP(INODP)
        UVELY=UVELY+VARB2(ITOTU)*SHAPE
        RADUS=RADUS+COORD(KPOIN,2)*SHAPE
        VVELY=VVELY+VARB2(ITOTV)*SHAPE
        SR11=SR11+VARB2(ITOTU)*CARTP(1,INODP)
        SR22=SR22+VARB2(ITOTV)*CARTP(2,INODP)
        SR12=SR12+.5*(VARB2(ITOTU)*CARTP(2,INODP)
     1  +VARB2(ITOTV)*CARTP(1,INODP))
50      continue
        SREF=SQRT(2*(SR11**2)+(SR12**2)+2*(SR22**2))
        IF (IITER.EQ.1) GOTO 51
        VSFAC=SREF**-.95
51      CONTINUE
        IF(IAXSY.EQ.1) DAREA=DAREA*RADUS
C
C       PUT BODY FORCES IN LOCAL RHS VECTOR
C
        DO 60 INODP=1,NNODP
        ERHSU(INODP)=ERHSU(INODP)+SHAPP(INODP)*DAREA*XFORC
        ERHSV(INODP)=ERHSV(INODP)+SHAPP(INODP)*DAREA*YFORC
60      CONTINUE
        DO 90 ICON1=1,4
        DO 90 ICON2=1,2
        IROWU=(ICON1-1)*5+3*ICON2-2
        IROWV=IROWU+3-ICON2
        INODP=2*(ICON1-1)+ICON2
```

```
      SHAPI=SHAPP(INODP)
      CARXI=CARTP(1,INODP)
      CARYI=CARTP(2,INODP)
      IROWP=IROWU+1
      DO 80 JCON1=1,4
      JCOLP=(JCON1-1)*5+2
C
C     PUT PRESSURE TERMS IN MOMENTUM EQUATIONS
C
      FLUMX(IROWU,JCOLP)=FLUMX(IROWU,JCOLP)+SHAPI*CARTL(1,JCON1)*
     1          DAREA/DENSY
      FLUMX(IROWV,JCOLP)=FLUMX(IROWV,JCOLP)+SHAPI*CARTL(2,JCON1)*
     1          DAREA/DENSY
      DO 80 JCON2=1,2
      JCOLU=(JCON1-1)*5+3*JCON2-2
      JCOLV=JCOLU+3-JCON2
      JNODP=2*(JCON1-1)+JCON2
      SHAPJ=SHAPP(JNODP)
      CARXJ=CARTP(1,JNODP)
      CARYJ=CARTP(2,JNODP)
C
C     PUT DIFFUSION & CONVECTION TERMS IN MOMENTUM & ENERGY  EQUATIONS
C
      DIFFU=(CARXI*CARXJ+CARYI*CARYJ)*VISCY*DAREA*VALT(IELEM,IGAUS)
     1 *VSFAC
      CONVC=(UVELY*CARXJ+VVELY*CARYJ)*SHAPI*DAREA
      FLUMX(IROWU,JCOLU)=FLUMX(IROWU,JCOLU)+DIFFU+CONVC
      FLUMX(IROWV,JCOLV)=FLUMX(IROWV,JCOLV)+DIFFU+CONVC
      IF(IAXSY.EQ.1) FLUMX(IROWV,JCOLV)=FLUMX(IROWV,JCOLV)+
     1       SHAPI*SHAPJ*VALT(IELEM,IGAUS)*VISCY*DAREA/RADUS**2
      IF (ICON2.EQ.2) GOTO 70
C
C     FORM CONTINUITY EQUATION
C
      FLUMX(IROWP,JCOLU)=FLUMX(IROWP,JCOLU)+SHAPL(ICON1)*DAREA*CARXJ
      FLUMX(IROWP,JCOLV)=FLUMX(IROWP,JCOLV)+SHAPL(ICON1)*DAREA*CARYJ
      IF(IAXSY.EQ.1) FLUMX(IROWP,JCOLV)=FLUMX(IROWP,JCOLV)+
     1       SHAPL(ICON1)*SHAPJ*DAREA/RADUS
70    CONTINUE
80    CONTINUE
90    CONTINUE
100   CONTINUE
C
C     ADD LOCAL RHS VECTOR TO GLOBAL ARRAY
C
      DO 110 INODP=1,NNODP
      KPOIN=IABS(LNODS(IELEM,INODP))
      ITOTU=NADFM(KPOIN)
      ITOTV=ITOTU+NODFM(KPOIN)-1
      EQRHS(ITOTU)=EQRHS(ITOTU)+ERHSU(INODP)
      EQRHS(ITOTV)=EQRHS(ITOTV)+ERHSV(INODP)
110   CONTINUE
      RETURN
      END
C
```

```
C
C
      SUBROUTINE TOLREL(IITER,MTOTV,NCONV,NTOTV,RELAX,TOLER,VARB1,
     1           VARB2,MPOIN)
      DIMENSION VARB1(MTOTV),VARB2(MTOTV),NADFM(MPOIN),
     1  NODFM(MPOIN)
C
C       CHECK TO SEE IF THE SOLUTIONS HAVE CONVERGED
C       CHECK FOR CONVERGENCE TO REQUIRED TOLERANCE
C
      CANLA=0
      NCONV=1
      DO 10 ITOTV=1,NTOTV
      IF(abs(VARB1(itotv)).LT..0001)GO TO 10
      CANGE=ABS((VARB1(ITOTV)-VARB2(ITOTV))/VARB1(itotv))
      IF(CANGE.GT.TOLER)NCONV=0
      IF(CANLA.GT.CANGE) GO TO 10
      CANLA=CANGE
      LTOLA=ITOTV
10    CONTINUE
      WRITE(7,2000)LTOLA,CANLA
      IF(NCONV.EQ.0) GO TO 20
      WRITE(7,2010)
      RETURN
20    CONTINUE
      IF(IITER.EQ.1)GO TO 40
C
C       RELAX VARIABLES FOR NEXT ITERATION
C
      DO 30 ITOTV=1,NTOTV
      VARB2(ITOTV)=VARB2(ITOTV)+RELAX*(VARB1(ITOTV)-VARB2(ITOTV))
30    CONTINUE
      RETURN
40    CONTINUE
      DO 50 ITOTV=1,NTOTV
      VARB2(ITOTV)=VARB1(ITOTV)
50    CONTINUE
2000  FORMAT(/37H LARGEST CHANGE OCCURS AT DOF NUMBER ,
     1      I4,10H CHANGE = ,F10.4)
2010  FORMAT(//45H SOLUTION HAS CONVERGED TO REQUIRED TOLERANCE)
      RETURN
      END
C
C
C
      SUBROUTINE ITERAT(BOUDV,COORD,DENSY,EQRHS,GFLUM,GRADB,IAXSY,
     1           LBOUD,LHEDV,LNODS,MELEM,MFRON,MPOIN,MTOTV,
     2           NADFM,NBCON,NDOFM,NELEM,NEVAB,NGAUS,NGRAD,
     3           NITER,NNODL,NNODP,NODFM,NPOIN,NTOTV,PNORM,
     4           POSGP,RELAX,TOLER,VARB1,VARB2,VISCY,WEIGP,
     5           XFORC,YFORC,VALT)
C
C       EXTERNAL SUBROUTINES:
C       PRESCR,FRONTS,WRITER,TOLREL
C
```

```
      DIMENSION EQRHS(MTOTV),NGRAD(MELEM),COORD(MPOIN,2),
     1  LNODS(MELEM,8)
     1  ,VARB1(MTOTV),VARB2(MTOTV),POSGP(3),WEIGP(3),BOUDV(MTOTV),
     2  GFLUM(MFRON,MFRON),LBOUD(MTOTV),LHEDV(MFRON),NODFM(MPOIN),
     3  PNORM(MFRON),NADFM(MPOIN),GRADB(MPOIN,4),VALT(MELEM,9)
C
C     SET UP ITERATION COUNTER AND LOOP ADDRESS
C
      IITER=0
10    CONTINUE
      IITER=IITER+1
      DO 20 ITOTV=1,NTOTV
      EQRHS(ITOTV)=0
20    CONTINUE
C
C     CALL PRESCR FOR ELEMENTS WITH NON-ZERO BOUNDARY GRADIENTS
C
      DO 30 IELEM=1,NELEM
      IF(NGRAD(IELEM).EQ.0)GO TO 30
      IELEMS=IELEM
      CALL PRESCR(COORD,EQRHS,GRADB,IAXSY,IELEMS,LNODS,MELEM,
     1    MPOIN,MTOTV,NADFM,NGAUS,NGRAD,NNODP,NODFM,
     2    POSGP,VISCY,WEIGP)
30    CONTINUE
C
C
C     CALL FRONTS TO SET UP AND SOLVE GOVERNING EQUATIONS
C
      CALL FRONTS(BOUDV,COORD,DENSY,EQRHS,GFLUM,IAXSY,IITER,
     1    LBOUD,LHEDV,LNODS,MELEM,MFRON,MPOIN,MTOTV,
     2    NADFM,NBCON,NELEM,NEVAB,NNODL,NNODP,NODFM,
     3    NPOIN,NTOTV,PNORM,VARB1,VARB2,VISCY,XFORC,
     4    YFORC,VALT)
C
C
C     CALL WRITER TO OUTPUT ITERATION RESULTS
C
      CALL WRITER(IITER,MPOIN,MTOTV,NADFM,NDOFM,NODFM,NPOIN,
     1    VARB1,VARB2)
C
C     CALL TOLREL TO CHECK CONVERGENCE. RELAX VALUES IF NOT CONVERGED.
C
      CALL TOLREL(IITER,MTOTV,NCONV,NTOTV,RELAX,TOLER,VARB1,VARB2)
C
C     RETURN TO MASTER NUMBER IF ITERATIONS EXCEED MAXIMUM
C
      IF(NCONV.EQ.1)RETURN
      IF(IITER.LT.NITER) GO TO 10
      WRITE(7,2000)
2000  FORMAT(//32H SOLUTION HAS FAILED TO CONVERGE)
      RETURN
      END
C
C
C
```

```
      SUBROUTINE FRONTS(BOUDV,COORD,DENSY,EQRHS,GFLUM,IAXSY,IITER,
     1            LBOUD,LHEDV,LNODS,MELEM,MFRON,MPOIN,MTOTV,
     2            NADFM,NBCON,NELEM,NEVAB,NNODL,NNODP,NODFM,
     3            NPOIN,NTOTV,PNORM,VARB1,VARB2,VISCY,XFORC,
     4            YFORC,VALT)
C
C       EXTERNAL SUBROUTINES:
C       MATRIX
C
      DIMENSION BOUDV(MTOTV),EQRHS(MTOTV),FLUMX(20,20),
     1  GFLUM(MFRON,MFRON),LBOUD(MTOTV),LHEDV(MFRON),
     2  LNODS(MELEM,8),LOCEL(20),NADFM(MPOIN),NDEST(20),
     3  NODFM(MPOIN),PNORM(MFRON),VARB1(MTOTV),VARB2(MTOTV)
     4  ,COORD(MPOIN,2),VALT(MELEM,9)
C
C       REWIND TAPES PRIOR TO SOLUTION PROCEDURE
C
      IF(IITER.GT.1)GO TO 40
C
C***    ON FIRST ITERATION ONLY FIND LAST APPEARANCE OF EACH NOD.
C
      DO 30 IPOIN=1,NPOIN
      LASTE=0
      DO 20 IELEM=1,NELEM
      DO 10 INODP=1,NNODP
      IF(LNODS(IELEM,INODP).NE.IPOIN)GO TO 10
      LASTE=IELEM
      LASTN=INODP
      GO TO 20
10    CONTINUE
20    CONTINUE
      LNODS(LASTE,LASTN)=-IPOIN
30    CONTINUE
40    CONTINUE
C
C***    INITIALISE HEADING AND GRAND FLUID MATRIX.
C
      REWIND 25
      REWIND 10
      NCRIT=MFRON-NEVAB
      NFRON=0
      DO 50 IFRON=1,MFRON
      DO 50 JFRON=1,MFRON
      GFLUM(IFRON,JFRON)=0.0
50    CONTINUE
      KELEM=0
C
C***    START  ASSEMBLY BY FORMING ELEMENTAL MATRICES
C
60    CONTINUE
      KELEM=KELEM+1
      CALL MATRIX(COORD,DENSY,EQRHS,FLUMX,IAXSY,LNODS,MELEM,
     1    MPOIN,MTOTV,NADFM,NEVAB,NNODL,NNODP,NODFM,
     2    VARB2,VISCY,XFORC,YFORC,VALT,IITER)
      KEVAB=0
```

```
C
C***     CREAT GLOBAL DOF ARRAY FOR EACH LOCAL ELEMENT DOF.
C
         DO 70 INODP=1,NNODP
         KPOIN=LNODS(KELEM,INODP)
         IADFM=NADFM(IABS(KPOIN))
         LODFM=NODFM(IABS(KPOIN))
         DO 70 IODFM=1,LODFM
         KEVAB=KEVAB+1
         LOCEL(KEVAB)=IADFM+IODFM-1
         IF(KPOIN.LT.0)LOCEL(KEVAB)=-LOCEL(KEVAB)
70       CONTINUE
C
C***     FIT EACH DOF INTO THE FRONT WIDTH EXTENDING IF NECESSARY.
C
         DO 120 IEVAB=1,NEVAB
         KTOTV=LOCEL(IEVAB)
         IF(NFRON.EQ.0)GO TO 90
         DO 80 IFRON=1,NFRON
         KFRON=IFRON
         IF(IABS(KTOTV).EQ.IABS(LHEDV(KFRON)))GO TO 110
80       CONTINUE
90       CONTINUE
         NFRON=NFRON+1
         IF(NFRON.LE.MFRON)GO TO 100
         WRITE(7, 2000)
         STOP
100      CONTINUE
         NDEST(IEVAB)=NFRON
         LHEDV(NFRON)=KTOTV
         GO TO 120
110      CONTINUE
         NDEST(IEVAB)=KFRON
         LHEDV(KFRON)=KTOTV
120      CONTINUE
C
C***     ASSEMBLE NEW ELEMENT INTO GRAND FLUID MATRIX.
C
         DO 130 IEVAB=1,NEVAB
         IFRON=NDEST(IEVAB)
         DO 130 JEVAB=1,NEVAB
         JFRON=NDEST(JEVAB)
         GFLUM(JFRON,IFRON)=GFLUM(JFRON,IFRON)+FLUMX(JEVAB,IEVAB)
130      CONTINUE
         IF(NFRON.LT.NCRIT.AND.KELEM.LT.NELEM)GO TO 60
140      CONTINUE
         NFSUM=0
         PIVOT=0.0
C
C***     CHECK LAST APPEARANCE OF EACH DOF PROCESS BOUNDRY CONDITIONS.
C
         DO 170 IFRON=1,NFRON
         IF(LHEDV(IFRON).GE.0)GO TO 170
         NFSUM=1
         IF(LBOUD(IABS(LHEDV(IFRON))).NE.1)GO TO 160
```

```
      IF(LPIVT.EQ.1)GO TO 270
      DO 260 JFRON=1,LPIVT-1
      GFLUM(IFRON-1,JFRON)=GFLUM(IFRON,JFRON)-FACOR*PNORM(JFRON)
260   CONTINUE
270   CONTINUE
      DO 280 JFRON=LPIVT+1,NFRON
      GFLUM(IFRON-1,JFRON-1)=GFLUM(IFRON,JFRON)-FACOR*PNORM(JFRON)
280   CONTINUE
      ITOTV=IABS(LHEDV(IFRON))
      EQRHS(ITOTV)=EQRHS(ITOTV)-FACOR*RHSID
290   CONTINUE
300   CONTINUE
C
C***  WRITE OUT NONFIXED PIVOTAL EQUATION ON TAPE.
C
      IF(LBOUD(KTOTV).NE.0)GO  TO 310
      WRITE(25)NFRON,LPIVT,(LHEDV(IFRON),PNORM(IFRON),IFRON=1,NFRON)
310   CONTINUE
      DO 320 IFRON=1,NFRON
      GFLUM(IFRON,NFRON)=0.0
      GFLUM(NFRON,IFRON)=0.0
320   CONTINUE
      IF(LPIVT.EQ.NFRON)GO TO 340
      DO 330 IFRON=LPIVT,NFRON-1
      LHEDV(IFRON)=LHEDV(IFRON+1)
330   CONTINUE
340   CONTINUE
      NFRON=NFRON-1
C
C***  ASSEMBLE ELIMINATE OR BACK SUBSTITUTE.
C
      IF(NFRON.GT.NCRIT)GO TO 140
      IF(KELEM.LT.NELEM)GO TO 60
      IF(NFRON.GT.0)GO TO 140
C
CC*** BACK SUBSTITUTION
C
      DO 350 ITOTV=1,NTOTV
      VARB1(ITOTV)=BOUDV(ITOTV)
      LBOUD(ITOTV)=-LBOUD(ITOTV)
350   CONTINUE
      DO 370 ITOTV=1,NTOTV-NBCON
      BACKSPACE 25
      READ(25)NFRON,LPIVT,(LHEDV(IFRON),PNORM(IFRON),IFRON=1,NFRON)
      KTOTV=IABS(LHEDV(LPIVT))
      TEMPR=0.0
      PNORM(LPIVT)=0.0
      DO 360 IFRON=1,NFRON
      TEMPR=TEMPR-PNORM(IFRON)*VARB1(IABS(LHEDV(IFRON)))
360   CONTINUE
      VARB1(KTOTV)=EQRHS(KTOTV)+TEMPR
      BACKSPACE 25
370   CONTINUE
      RETURN
2000  FORMAT(//39H PROGRAM HALTED FRONTWIDTH IS TOO SMALL)
```

```
      END
C
C
C
      SUBROUTINE DINPUT(BOUDV,COORD,DENSY,GRADB,IAXSY,LBOUD,LNODS,
     1            MELEM,MPOIN,MTOTV,NADFM,NBCON,NDOFM,NELEM,
     2            NEVAB,NGAUS,NGRAD,NITER,NNODL,NNODP,NODFM,
     3            NPOIN,NTOTV,RELAX,TOLER,VARB1,VARB2,VISCY,
     4            XFORC,YFORC,SLOPE1,D,E,FAC,XMAX,YMAX)
      DIMENSION BOUDV(MTOTV),COORD(MPOIN,2),GRADB(MPOIN,4),
     1  LBOUD(MTOTV),LNODS(MELEM,8),NADFM(MPOIN),
     2  NGRAD(MELEM),NODFM(MPOIN),TITLE(20),VARB1(MTOTV),
     3  VARB2(MTOTV)
C
C     GIVE VALUES TO THOSE VARIABLES WHICH CANNOT BE CHANGED
C
      NDOFM=3
      NEVAB=20
      NGAUS=3
      NNODL=4
      NNODP=8
C
C     READ IN EACH LINE AND ECHO IMMEDIATELY
C
      READ(40,1000)TITLE
      WRITE(7,2000)TITLE
      READ(40,*) IAXSY,NITER,NEX,NEY,XMAX,YMAX,angle
      WRITE(7,2015)IAXSY,NITER,NEX,NEY,XMAX,YMAX,angle
C     READ(40,*)IAXSY,NELEM,NITER,NPOIN,NRPON
C     WRITE(7,2010)IAXSY,NELEM,NITER,NPOIN,NRPON
      READ(40,*)NICON,NBCON,NEBCN,NNBCN
      WRITE(7,2020)NICON,NBCON,NEBCN,NNBCN
      CALL MESang(XMAX,YMAX,NEX,NEY,LNODS,COORD,NELEM,NPOIN,MELEM
     1  ,mpoin,angle)
C
C     CHECK INITIAL DATA
C
      CALL DIAGN1(NBCON,NEBCN,NELEM,NICON,NNBCN,NPOIN,NRPON)
C
C     READ FLOW PARAMETERS ETC.
C
      READ(40,*) DENSY,RELAX,TOLER,VISCY,XFORC,YFORC
      WRITE(7,2030)DENSY,RELAX,TOLER,VISCY,XFORC,YFORC
      READ(40,*) SLOPE1,E,D,FAC
      GOTO 25
C
C     ZERO ALL COORDINATE ARRAY
C
      DO 10 IPOIN=1,NPOIN
      DO 10 IDIME=1,2
      COORD(IPOIN,IDIME)=0
10    CONTINUE
C
C     READ IN NODAL COORDINATES (X THEN Y)
C
```

```
      DO 20 IPOIN=1,NRPON
      READ(40,*) JPOIN,(COORD(JPOIN,IDIME),IDIME=1,2)
20    CONTINUE
C
C     READ IN NODAL CONNECTIONS AND ECHO IMMEDIATELY
C
25    CONTINUE
      WRITE(7,2040)
      DO 30 IELEM=1,NELEM
      NGRAD(IELEM)=0
C     READ(40,*) IELEM,(LNODS(IELEM,INODP),INODP=1,NNODP)
      WRITE(7,2050)IELEM,(LNODS(IELEM,INODP),INODP=1,NNODP)
30    CONTINUE
C
C     SET UP VECTOR GIVING THE D.O.F. AT EACH NODE
C
      ITEMP=NNODP-1
      DO 40 IELEM=1,NELEM
      DO 40 INODP=1,ITEMP,2
      NODFM(LNODS(IELEM,INODP))=NDOFM
      NODFM(LNODS(IELEM,INODP+1))=NDOFM-1
40    CONTINUE
      GOTO 71
C
C     INTERPOLATE MIDSIDE NODE COORDINATES WHERE NOT GIVEN
C
      DO 70 IELEM=1,NELEM
      DO 60 INODP=2,NNODP,2
      IPOIN=LNODS(IELEM,INODP)
      TEMPY=ABS(COORD(IPOIN,1))+ABS(COORD(IPOIN,2))
      IF(TEMPY.NE.0.0) GO TO 60
      JPOIN=LNODS(IELEM,INODP-1)
      KNODP=INODP+1
      IF(KNODP.GT.NNODP)KNODP=1
      KPOIN=LNODS(IELEM,KNODP)
      DO 50 IDIME=1,2
      COORD(IPOIN,IDIME)=(COORD(JPOIN,IDIME)+COORD(KPOIN,IDIME))*.5
50    CONTINUE
60    CONTINUE
70    CONTINUE
71    CONTINUE
C
C     SET UP VECTOR WITH THE FIRST D.O.F. AT EACH NODE
C
      NADFM(1)=1
      DO 80 IPOIN=2,NPOIN
      NADFM(IPOIN)=NADFM(IPOIN-1)+NODFM(IPOIN-1)
80    CONTINUE
      NTOTV=NADFM(NPOIN)+NODFM(NPOIN)-1
C
C     INITIALIZE REMAINING ARRAYS
C
      DO 90 ITOTV=1,NTOTV
      LBOUD(ITOTV)=0
      BOUDV(ITOTV)=0
```

```
      VARB1(ITOTV)=0
      VARB2(ITOTV)=0
90    CONTINUE
C
C     READ IN INITIAL CONDITIONS(GUESS VALUES) IF ANY
C
      IF(NICON.EQ.0) GO TO 110
      DO 100 IICON=1,NICON
      READ(40,*)IPOIN,IDOFM,TEMPY
      IF(IDOFM.GT.1)IDOFM=NODFM(IPOIN)-1
      JTOTV=NADFM(IPOIN)+IDOFM-1
      VARB1(JTOTV)=TEMPY
      VARB2(JTOTV)=TEMPY
100   CONTINUE
110   CONTINUE
C
C     WRITE OUT COORDINATES AND ARRAYS SHOWING DOF
C
      WRITE(7,2060)
      DO 130 IPOIN=1,NPOIN
      IF(NODFM(IPOIN).NE.NDOFM) GO TO 120
      WRITE(7,2070)IPOIN,NADFM(IPOIN),NODFM(IPOIN),
     1 (COORD(IPOIN,IDIME),IDIME=1,2),
     2 (VARB1(NADFM(IPOIN)-1+IODFM),IODFM=1,NODFM(IPOIN))
      GO TO 130
120   CONTINUE
      WRITE(7,2080)IPOIN,NADFM(IPOIN),NODFM(IPOIN),
     1 (COORD(IPOIN,IDIME),IDIME=1,2),
     2 (VARB1(NADFM(IPOIN)-1+IODFM),IODFM=1,NODFM(IPOIN))
130   CONTINUE
C
C     CHECK NODAL CONNECTIONS AND COORDINATES
C
      CALL DIAGN2(COORD,DENSY,LNODS,MELEM,MPOIN,NELEM,NICON,
     1    NNODP,NPOIN,NTOTV,VISCY)
C
C     READ IN BOUNDARY CONDITIONS
C
      WRITE(7,2090)
      DO 170 IBCON=1,NBCON
      IF (NBCON.EQ.0) GOTO 171
      READ(40,*)IPOIN,IDOFM,BVALU
      ITOTV=NADFM(IPOIN)+IDOFM-1
      IF(IDOFM.GT.NODFM(IPOIN))ITOTV=ITOTV-1
      LBOUD(ITOTV)=1
      BOUDV(ITOTV)=BVALU
      GO TO (140,150,160),IDOFM
140   CONTINUE
      WRITE(7,2100)IPOIN,BVALU
      GO TO 170
150   CONTINUE
      WRITE(7,2110)IPOIN,BVALU
      IF(NODFM(IPOIN).NE.IDOFM) GO TO 170
      WRITE(7,2120)IBCON,IPOIN
      STOP
```

```
160       CONTINUE
          WRITE(7,2130)IPOIN,BVALU
170       CONTINUE
171       CONTINUE
C


C         READ IN ELEMENTS AND SIDES WITH GRADIENT BOUNDARY CONDITIONS
C
          IF(NEBCN.EQ.0) RETURN
          WRITE(7,2140)
          DO 180 IEBCN=1,NEBCN
          READ(40,*)IELEM,NGRAD(IELEM)
          WRITE(7,2150)IELEM,NGRAD(IELEM)
180       CONTINUE
          WRITE(7,2160)
          DO 190 IPOIN=1,NPOIN
          DO 190 IGRAD=1,4
          GRADB(IPOIN,IGRAD)=0
190       CONTINUE
C
C         READ IN VALUES OF THE GRADIENTS ON BOUNDARY NODES
C
          DO 240 INBCN=1,NNBCN
          READ(40,*)IPOIN,IGRAD,GRADB(IPOIN,IGRAD)
          GO TO (200,210,220,230),IGRAD
200       CONTINUE
          WRITE(7,2100)IPOIN,GRADB(IPOIN,IGRAD)
          GO TO  240
210       CONTINUE
          WRITE(7,2110)IPOIN,GRADB(IPOIN,IGRAD)
          GO TO 240
220       CONTINUE
          WRITE(7,2130)IPOIN,GRADB(IPOIN,IGRAD)
          GO TO 240
230       CONTINUE
          WRITE(7,2170)IPOIN,GRADB(IPOIN,IGRAD)
240       CONTINUE
          RETURN
1000      FORMAT(80A1)
2000      FORMAT(14H OUTPUT FILE  :,//1X,20A4)
2010      FORMAT(//13H CONTROL DATA,/13H ************,//
     18H IAXSY =,I4,4X,8H NELEM =,I4,4X,8H NITER =,I4,4X,
     28H NPOIN =,I4,4X,8H NRPON =,I4)
2015      FORMAT(//13H CONTROL DATA,/13H ************,//
     18H IAXSY =,I4,6X,8H NITER =,I4,6X,8H   NEX =,I4,6X,
     28H   NEY =,I4,/8H  XMAX =,F6.2,4X,8H  YMAX =,F6.2,4X,
     38H angle =,f7.2)
2020      FORMAT(8H NICON =,I4,6X,8H NBCON =,
     1I4,6X,8H NEBCN =,I4,6X,8H NNBCN =,I4)
2030      FORMAT(//20H PHYSICAL PROPERTIES,/20H ******************,//
     18H DENSY =,F10.5,4X,8H RELAX =,F10.5,4X,8H TOLER =,
     2F10.5,/8H VISCY =,F10.5,4X,8H XFORC =,F10.5,4X,
     38H YFORC =,F10.5)
2040      FORMAT(//17H ELEMENT TOPOLOGY,/17H ***************,//
```

```
     18H ELEMENT,6X,12HNODE NUMBERS)
2050    FORMAT(1X,I5,6X,8I5)
2060    FORMAT(//16H NODE POINT DATA,/16H ***************,//
     134H NODE NADFM NODFM X-COORD   Y-COORD,3X,
     225HU-VEL     PRESSURE    V-VEL)
2070    FORMAT(3(I4,1X),5(F9.4,1X))
2080    FORMAT(3(I4,1X),3(F9.4,1X),10X,F9.4)
2090    FORMAT(//20H BOUNDARY CONDITIONS,/20H ********************,//
     135H NODE   U-FIXED  PRESSURE   V-FIXED    )
2100    FORMAT(I5,F9.4)
2110    FORMAT(I5,10X,F9.4)
2120    FORMAT(/27H ERROR!! BOUNDARY CONDITION,I4,
     142H DEFINES A PRESSURE AT MIDSIDE NODE NUMBER,I4)
2130    FORMAT(I5,20X,F9.4)
2140    FORMAT(/25H VELOCITY GRADIENT VALUES,/,1X,24('*'),
     1//14H ELEMENT  SIDE)
2150    FORMAT(I5,3X,I5)
2160    FORMAT(//44H NODE     DU/DX      DU/DY      DV/DX      DV/DY)
2170    FORMAT(I5,30X,F9.4)
        END
C
C
C
        SUBROUTINE DIAGN1(NBCON,NEBCN,NELEM,NICON,NNBCN,NPOIN,NRPON)
        DIMENSION NECHO(80),NEROR(8)
C
C       SCRUTINY OF CONTROL DATA
C
        DO 10 IEROR=1,8
10      NEROR(IEROR)=0
C
C       SCRUTINIZE CONTROL DATA AND PRINT ERROR MESSAGE
C
        IF(NPOIN.LT.0)                NEROR(1)=1
C       COMMENT NEXT LINE IF NOT READING NRPON.
C       IF(NRPON.LE.0.OR.NRPON.GT.NPOIN)  NEROR(2)=1
        IF(NELEM.LE.0)                NEROR(3)=1
        IF(NPOIN.GT.NELEM*8)            NEROR(4)=1
        IF(NBCON.GT.NPOIN*3)            NEROR(5)=1
        IF(NICON.GT.NPOIN*3)            NEROR(6)=1
        IF(NEBCN.GT.NELEM)                NEROR(7)=1
        IF(NNBCN.GT.NPOIN*4)            NEROR(8)=1
C
C       CHECK ON ERROR. IF UNITY, PRINT.
C
        JEROR=0
        DO 20 IEROR=1,8
        IF(NEROR(IEROR).EQ.0)GO TO 20
        JEROR=1
C
C       WRITE OUT ERROR NUMBER
C
        WRITE(7,2000) IEROR
20      CONTINUE
        IF(JEROR.EQ.0) RETURN
```

```
C
C       LIST DATA REMAINING AFTER CONTROL PARAMETERS
C
        WRITE(7,2010)
30      CONTINUE
        READ(40,1000)NECHO
        WRITE(7,2020)NECHO
        GO TO 30
1000    FORMAT(80A1)
2000    FORMAT(/,10X,33HCONTROL PARAMETER ERROR*****ERROR,I5)
2010    FORMAT(/10X,37HDATA FOLLOWING ERROR IN CONTROL CARDS/)
2020    FORMAT(10X,80A1)
        END
C
C
C
        SUBROUTINE DIAGN2(COORD,DENSY,LNODS,MELEM,MPOIN,NELEM,NICON,
     1            NNODP,NPOIN,NTOTV,VISCY)
        DIMENSION COORD(MPOIN,2),LNODS(MELEM,8),NECHO(80),NEROR(13)
C
C       SCRUTINIZE ELEMENT AND NODAL POINT DATA
C       INITIALIZE ERROR ARRAY
C
        DO 10 IEROR=1,13
        NEROR(IEROR)=0
10      CONTINUE
C
C       CHECK PHYSICAL PROPERTIES
C
        IF(DENSY.LE.0.OR.VISCY.LE.0) NEROR(9)=1
C
C       CHECK NODAL COORDINATES
C       CHECK FOR TWO IDENTICAL COORDINATES
C
        DO 40 IPOIN=2,NPOIN
        JPOIN=IPOIN-1
        DO 30 KPOIN=1,JPOIN
        IF(COORD(IPOIN,1).NE.COORD(KPOIN,1).OR.COORD(IPOIN,2).
     1   NE.COORD(KPOIN,2)) GO TO 20
        NEROR(10)=1
20      CONTINUE
30      CONTINUE
40      CONTINUE
C
C       CHECK ELEMENT NODAL NUMBERS
C       A - REPETITION OF A NODE NUMBER IN ONE ELEMENT
C       B - NODE NUMBER OUTSIDE PERMISSIBLE BOUNDS
C
        DO 50 IPOIN=1,NPOIN
        DO 50 IELEM=1,NELEM
        LCONT=0
        DO 50 INODE=1,NNODP
        IF(LNODS(IELEM,INODE).NE.IPOIN) GO TO 50
        LCONT=LCONT+1
        IF(LCONT.GT.1) NEROR(11)=1
```

```
50      CONTINUE
        DO 60 IELEM=1,NELEM
        DO 60 INODE=1,NNODP
        IF(LNODS(IELEM,INODE).LE.0.OR.LNODS(IELEM,INODE).GT.NPOIN)
     1NEROR(12)=1
60      CONTINUE
C
C       CHECK ON NUMBER OF INITIAL CONDITIONS INPUT
C
        IF(NICON.GT.NTOTV) NEROR(13)=1
C
C       WRITE OUT ERROR NUMBER
C
        JEROR=0
        DO 70 IEROR=9,13
        IF(NEROR(IEROR).EQ.0) GO TO 70
        JEROR=1
        WRITE(7,2000) IEROR
70      CONTINUE
C
C       DECIDE WHETHER TO CONTINUE OR ECHO DATA
C
        IF(JEROR.EQ.0) RETURN
C
C       LIST REMAINING DATA
C
        WRITE(7,2010)
80      CONTINUE
        READ(40,1000) NECHO
        WRITE(7,2020) NECHO
        GO TO 80
1000    FORMAT(80A1)
2000    FORMAT(/10X,10HDATA ERROR,I5)
2010    FORMAT(/10X,14HREMAINING DATA/)
2020    FORMAT(10X,80A1)
        END
C
        SUBROUTINE MATPRT(COORD,VALT,LNODS,MELEM,MPOIN,NELEM,NGAUS,
     1  POSGP,XMAX,YMAX,SLOPE1,D,E,FAC,NNODP)
        DIMENSION COORD(MPOIN,2),VALT(MELEM,9),POSGP(3),
     1  SHAPE(8),LNODS(MELEM,8),DERIV(2,8)
        LGAUS=NGAUS*NGAUS
        DO 11 IELEM=1,NELEM
        DO 10 KGAUS=1,LGAUS
        VALT(IELEM,KGAUS)=1.0
10      CONTINUE
11      CONTINUE
        DO 100 IELEM=1,NELEM
        KGAUS=0
        DO 90 IGAUS=1,NGAUS
        DO 90 JGAUS=1,NGAUS
        KGAUS=KGAUS+1

C       FINDING COORDINATES OF GAUSS-POINTS
        XEQIV=POSGP(IGAUS)
```

```
      CALL SHAPE8(DERIV,SHAPE,XEQIV,YEQIV)
      XC=0.0
      YC=0.0
c     shape-func=1,8
      DO 80 INODP=1,NNODP
      KPOIN=IABS(LNODS(IELEM,INODP))
      XC=XC+SHAPE(INODP)*COORD(KPOIN,1)
      YC=YC+SHAPE(INODP)*COORD(KPOIN,2)
80    CONTINUE
C     CHECK THE POSITION OF GAUSS-POINTS
      YCRU=SLOPE1*(XC-XMAX/2)+(YMAX/2+D*SQRT(1+SLOPE1**2))
      YCRL=SLOPE1*(XC-XMAX/2)+(YMAX/2-D*SQRT(1+SLOPE1**2))
      SLOPE2=-1/SLOPE1
      XCRU=(YC-YMAX/2)/SLOPE2+(XMAX/2+E*SQRT(1+SLOPE1**2))
      XCRL=(YC-YMAX/2)/SLOPE2+(XMAX/2-E*SQRT(1+SLOPE1**2))
      IF ((XC.LE.XCRU).AND.(XC.GE.XCRL).AND.(YC.LE.YCRU).
     1  AND.(YC.GE.YCRL)) THEN
      VALT(IELEM,KGAUS)=FAC
      ELSE
      ENDIF
90    CONTINUE
100   CONTINUE
      DO 110 IELEM=1,NELEM
      WRITE(7,*) IELEM,(VALT(IELEM,KGAUS),KGAUS=1,LGAUS)
110   CONTINUE
      RETURN
      END

C
C
      SUBROUTINE WRITER(IITER,MPOIN,MTOTV,NADFM,NDOFM,NODFM,NPOIN,
     1          VARB1,VARB2)
      DIMENSION NADFM(MPOIN),NODFM(MPOIN),VARB1(MTOTV),
     2  VARB2(MTOTV)
      WRITE(7,2000)IITER
      DO 20 IPOIN=1,NPOIN
      IODFM=NODFM(IPOIN)
      IADFM=NADFM(IPOIN)
      IF(IODFM.EQ.NDOFM) GO TO 10
c     WRITE(7,2010)IPOIN,(VARB1(IADFM+JODFM-1),JODFM=1,IODFM),
c    1            (VARB2(IADFM+JODFM-1),JODFM=1,IODFM)
      WRITE(7,2010)IPOIN,(VARB1(IADFM+JODFM-1),JODFM=1,IODFM)
      GO TO 20
10    CONTINUE
C     WRITE(7,2020)IPOIN,(VARB1(IADFM+JODFM-1),JODFM=1,IODFM),
C    1            (VARB2(IADFM+JODFM-1),JODFM=1,IODFM)
      WRITE(7,2020)IPOIN,(VARB1(IADFM+JODFM-1),JODFM=1,IODFM)
20    CONTINUE
      RETURN
2000  FORMAT(//29H RESULTS FOR ITERATION NUMBER,I4,
     1 //14X,3(3HNEW,7X),4X,3(3HOLD,7X)/,
     27H  NODE ,2(4X,30HU-VELOCITY PRESSURE V-VELOCITY))
2010  FORMAT(I6,4X,(E10.3,10X,E10.3,5X))
2020  FORMAT(I6,4X,(3E10.3,5X))
```

```
C
C
      SUBROUTINE MESang(XMAX,YMAX,NEX,NEY,LNODS,COORD,NELEM,NPOIN,
     1          MELEM,MPOIN,angle)
      DIMENSION LNODS(MELEM,8),COORD(MPOIN,2),b(mpoin)
      DX=XMAX/(2*NEX)
      DY=YMAX/(2*NEY)
      NPOIN=(2*NEX+1)*(2*NEY+1)-NEX*NEY
      NELEM=NEX*NEY
      a=ymax/tand(angle)
      if(2*a.lt.xmax) goto 5
      write(5,1000)
1000  format(//10x,'incompatible mesh geometry')
      stop
5     dxt=(xmax-2*a)/(2*nex)
      ddx=dx-dxt
      do 25 ipoin=1,npoin
      b(ipoin)=0
25    continue
      DO 30 IELEM=1,NELEM
      J1=IELEM/NEX
      IF(IELEM.EQ.J1*NEX) GOTO 10
      J=J1+1
      GOTO 20
10    CONTINUE
      J=J1
20    CONTINUE
      I=IELEM-(J-1)*NEX
      N1=(3*NEX+2)*(J-1)+2*(I-1)+1
      LNODS(IELEM,1)=N1
      LNODS(IELEM,2)=N1+1
      LNODS(IELEM,3)=N1+2
      LNODS(IELEM,4)=2*NEX-I+N1+3
      LNODS(IELEM,5)=3*NEX+N1+4
      LNODS(IELEM,6)=LNODS(IELEM,5)-1
      LNODS(IELEM,7)=LNODS(IELEM,6)-1
      LNODS(IELEM,8)=LNODS(IELEM,4)-1
      X1=(I-1)*2*DX
      Y1=(J-1)*2*DY
      COORD(LNODS(IELEM,1),1)=X1
      COORD(LNODS(IELEM,2),1)=X1+DX
      COORD(LNODS(IELEM,3),1)=X1+2*DX
      COORD(LNODS(IELEM,4),1)=X1+2*DX
      COORD(LNODS(IELEM,5),1)=X1+2*DX
      COORD(LNODS(IELEM,6),1)=X1+DX
      COORD(LNODS(IELEM,7),1)=X1
      COORD(LNODS(IELEM,8),1)=X1
      COORD(LNODS(IELEM,1),2)=Y1
      COORD(LNODS(IELEM,2),2)=Y1
      COORD(LNODS(IELEM,3),2)=Y1
      COORD(LNODS(IELEM,4),2)=Y1+DY
      COORD(LNODS(IELEM,5),2)=Y1+2*DY
      COORD(LNODS(IELEM,6),2)=Y1+2*DY
      COORD(LNODS(IELEM,7),2)=Y1+2*DY
```

```
      tan1=ymax/(a-2*(i-1)*ddx)
      tan2=ymax/(a-(2*i-1)*ddx)
      tan3=ymax/(a-2*i*ddx)
      b(lnods(ielem,1))=coord(lnods(ielem,1),2)/tan1
      b(lnods(ielem,2))=coord(lnods(ielem,2),2)/tan2
      b(lnods(ielem,3))=coord(lnods(ielem,3),2)/tan3
      b(lnods(ielem,4))=coord(lnods(ielem,4),2)/tan3
      b(lnods(ielem,5))=coord(lnods(ielem,5),2)/tan3
      b(lnods(ielem,6))=coord(lnods(ielem,6),2)/tan2
      b(lnods(ielem,7))=coord(lnods(ielem,7),2)/tan1
      b(lnods(ielem,8))=coord(lnods(ielem,8),2)/tan1
30    CONTINUE
      do 40 ipoin=1,npoin
      coord(ipoin,1)=coord(ipoin,1)+b(ipoin)
40    continue
      RETURN
      END
```

Appendix C

OUTPUT FILE

a 0 . 5 v f = . 4 4 1 / d = 1 . 5

CONTROL DATA
****

IAXSY =	0	NITER =	4	NEX =	8	NEY =	8		
XMAX =	1.00	YMAX =	1.00	angle =	90.00				
NICON =	0	NBCON =	86	NEBCN =	0	NNBCN =	0		

PHYSICAL PROPERTIES
****

DENSY =	1.00000	RELAX =	.50000	TOLER =	.01000
VISCY =	100.00000	XFORC =	.00000	YFORC =	.00000

ELEMENT TOPOLOGY
****

ELEMENT	NODE NUMBERS							
1	1	2	3	19	29	28	27	18
2	3	4	5	20	31	30	29	19
3	5	6	7	21	33	32	31	20
4	7	8	9	22	35	34	33	21
5	9	10	11	23	37	36	35	22
6	11	12	13	24	39	38	37	23
7	13	14	15	25	41	40	39	24
8	15	16	17	26	43	42	41	25
9	27	28	29	45	55	54	53	44
10	29	30	31	46	57	56	55	45
11	31	32	33	47	59	58	57	46
12	33	34	35	48	61	60	59	47
13	35	36	37	49	63	62	61	48
14	37	38	39	50	65	64	63	49
15	39	40	41	51	67	66	65	50
16	41	42	43	52	69	68	67	51
17	53	54	55	71	81	80	79	70
18	55	56	57	72	83	82	81	71
19	57	58	59	73	85	84	83	72
20	59	60	61	74	87	86	85	73
21	61	62	63	75	89	88	87	74
22	63	64	65	76	91	90	89	75
23	65	66	67	77	93	92	91	76
24	67	68	69	78	95	94	93	77
25	79	80	81	97	107	106	105	96
26	81	82	83	98	109	108	107	97
27	83	84	85	99	111	110	109	98
28	85	86	87	100	113	112	111	99
29	87	88	89	101	115	114	113	100
30	89	90	91	102	117	116	115	101
31	91	92	93	103	119	118	117	102
32	93	94	95	104	121	120	119	103
33	105	106	107	123	133	132	131	122
34	107	108	109	124	135	134	133	123
35	109	110	111	125	137	136	135	124
36	111	112	113	126	139	138	137	125
37	113	114	115	127	141	140	139	126
38	115	116	117	128	143	142	141	127
39	117	118	119	129	145	144	143	128
40	119	120	121	130	147	146	145	129

41	131	132	133	149	159	158	157	148
42	133	134	135	150	161	160	159	149
43	135	136	137	151	163	162	161	150
44	137	138	139	152	165	164	163	151
45	139	140	141	153	167	166	165	152
46	141	142	143	154	169	168	167	153
47	143	144	145	155	171	170	169	154
48	145	146	147	156	173	172	171	155
49	157	158	159	175	185	184	183	174
50	159	160	161	176	187	186	185	175
51	161	162	163	177	189	188	187	176
52	163	164	165	178	191	190	189	177
53	165	166	167	179	193	192	191	178
54	167	168	169	180	195	194	193	179
55	169	170	171	181	197	196	195	180
56	171	172	173	182	199	198	197	181
57	183	184	185	201	211	210	209	200
58	185	186	187	202	213	212	211	201
59	187	188	189	203	215	214	213	202
60	189	190	191	204	217	216	215	203
61	191	192	193	205	219	218	217	204
62	193	194	195	206	221	220	219	205
63	195	196	197	207	223	222	221	206
64	197	198	199	208	225	224	223	207

NODE POINT DATA
****

NODE	NADFM	NODFM	X-COORD	Y-COORD	U-VEL	P	V-VEL
1	1	3	.0000	.0000	.0000	.0000	.0000
2	4	2	.0625	.0000	.0000		.0000
3	6	3	.1250	.0000	.0000	.0000	.0000
4	9	2	.1875	.0000	.0000		.0000
5	11	3	.2500	.0000	.0000	.0000	.0000
6	14	2	.3125	.0000	.0000		.0000
7	16	3	.3750	.0000	.0000	.0000	.0000
8	19	2	.4375	.0000	.0000		.0000
9	21	3	.5000	.0000	.0000	.0000	.0000
10	24	2	.5625	.0000	.0000		.0000
11	26	3	.6250	.0000	.0000	.0000	.0000
12	29	2	.6875	.0000	.0000		.0000
13	31	3	.7500	.0000	.0000	.0000	.0000
14	34	2	.8125	.0000	.0000		.0000
15	36	3	.8750	.0000	.0000	.0000	.0000
16	39	2	.9375	.0000	.0000		.0000
17	41	3	1.0000	.0000	.0000	.0000	.0000
18	44	2	.0000	.0625	.0000		.0000
19	46	2	.1250	.0625	.0000		.0000
20	48	2	.2500	.0625	.0000		.0000
21	50	2	.3750	.0625	.0000		.0000
22	52	2	.5000	.0625	.0000		.0000
23	54	2	.6250	.0625	.0000		.0000
24	56	2	.7500	.0625	.0000		.0000
25	58	2	.8750	.0625	.0000		.0000
26	60	2	1.0000	.0625	.0000		.0000
27	62	3	.0000	.1250	.0000	.0000	.0000
28	65	2	.0625	.1250	.0000		.0000
29	67	3	.1250	.1250	.0000	.0000	.0000
30	70	2	.1875	.1250	.0000		.0000
31	72	3	.2500	.1250	.0000	.0000	.0000
32	75	2	.3125	.1250	.0000		.0000
33	77	3	.3750	.1250	.0000	.0000	.0000
34	80	2	.4375	.1250	.0000		.0000
35	82	3	.5000	.1250	.0000	.0000	.0000
36	85	2	.5625	.1250	.0000		.0000

37	87	3	.6250	.1250	.0000	.0000	.0000
38	90	2	.6875	.1250	.0000		.0000
39	92	3	.7500	.1250	.0000	.0000	.0000
40	95	2	.8125	.1250	.0000		.0000
41	97	3	.8750	.1250	.0000	.0000	.0000
42	100	2	.9375	.1250	.0000		.0000
43	102	3	1.0000	.1250	.0000	.0000	.0000
44	105	2	.0000	.1875	.0000		.0000
45	107	2	.1250	.1875	.0000		.0000
46	109	2	.2500	.1875	.0000		.0000
47	111	2	.3750	.1875	.0000		.0000
48	113	2	.5000	.1875	.0000		.0000
49	115	2	.6250	.1875	.0000		.0000
50	117	2	.7500	.1875	.0000		.0000
51	119	2	.8750	.1875	.0000		.0000
52	121	2	1.0000	.1875	.0000		.0000
53	123	3	.0000	.2500	.0000	.0000	.0000
54	126	2	.0625	.2500	.0000		.0000
55	128	3	.1250	.2500	.0000	.0000	.0000
56	131	2	.1875	.2500	.0000		.0000
57	133	3	.2500	.2500	.0000	.0000	.0000
58	136	2	.3125	.2500	.0000		.0000
59	138	3	.3750	.2500	.0000	.0000	.0000
60	141	2	.4375	.2500	.0000		.0000
61	143	3	.5000	.2500	.0000	.0000	.0000
62	146	2	.5625	.2500	.0000		.0000
63	148	3	.6250	.2500	.0000	.0000	.0000
64	151	2	.6875	.2500	.0000		.0000
65	153	3	.7500	.2500	.0000	.0000	.0000
66	156	2	.8125	.2500	.0000		.0000
67	158	3	.8750	.2500	.0000	.0000	.0000
68	161	2	.9375	.2500	.0000		.0000
69	163	3	1.0000	.2500	.0000	.0000	.0000
70	166	2	.0000	.3125	.0000		.0000
71	168	2	.1250	.3125	.0000		.0000
72	170	2	.2500	.3125	.0000		.0000
73	172	2	.3750	.3125	.0000		.0000
74	174	2	.5000	.3125	.0000		.0000
75	176	2	.6250	.3125	.0000		.0000
76	178	2	.7500	.3125	.0000		.0000
77	180	2	.8750	.3125	.0000		.0000
78	182	2	1.0000	.3125	.0000		.0000
79	184	3	.0000	.3750	.0000	.0000	.0000
80	187	2	.0625	.3750	.0000		.0000
81	189	3	.1250	.3750	.0000	.0000	.0000
82	192	2	.1875	.3750	.0000		.0000
83	194	3	.2500	.3750	.0000	.0000	.0000
84	197	2	.3125	.3750	.0000		.0000
85	199	3	.3750	.3750	.0000	.0000	.0000
86	202	2	.4375	.3750	.0000		.0000
87	204	3	.5000	.3750	.0000	.0000	.0000
88	207	2	.5625	.3750	.0000		.0000
89	209	3	.6250	.3750	.0000	.0000	.0000
90	212	2	.6875	.3750	.0000		.0000
91	214	3	.7500	.3750	.0000	.0000	.0000
92	217	2	.8125	.3750	.0000		.0000
93	219	3	.8750	.3750	.0000	.0000	.0000
94	222	2	.9375	.3750	.0000		.0000
95	224	3	1.0000	.3750	.0000	.0000	.0000
96	227	2	.0000	.4375	.0000		.0000
97	229	2	.1250	.4375	.0000		.0000
98	231	2	.2500	.4375	.0000		.0000
99	233	2	.3750	.4375	.0000		.0000
100	235	2	.5000	.4375	.0000		.0000
101	237	2	.6250	.4375	.0000		.0000
102	239	2	.7500	.4375	.0000		.0000

103	241	2	.8750	.4375	.0000		.0000
104	243	2	1.0000	.4375	.0000		.0000
105	245	3	.0000	.5000	.0000	.0000	.0000
106	248	2	.0625	.5000	.0000		.0000
107	250	3	.1250	.5000	.0000	.0000	.0000
108	253	2	.1875	.5000	.0000		.0000
109	255	3	.2500	.5000	.0000	.0000	.0000
110	258	2	.3125	.5000	.0000		.0000
111	260	3	.3750	.5000	.0000	.0000	.0000
112	263	2	.4375	.5000	.0000		.0000
113	265	3	.5000	.5000	.0000	.0000	.0000
114	268	2	.5625	.5000	.0000		.0000
115	270	3	.6250	.5000	.0000	.0000	.0000
116	273	2	.6875	.5000	.0000		.0000
117	275	3	.7500	.5000	.0000	.0000	.0000
118	278	2	.8125	.5000	.0000		.0000
119	280	3	.8750	.5000	.0000	.0000	.0000
120	283	2	.9375	.5000	.0000		.0000
121	285	3	1.0000	.5000	.0000	.0000	.0000
122	288	2	.0000	.5625	.0000		.0000
123	290	2	.1250	.5625	.0000		.0000
124	292	2	.2500	.5625	.0000		.0000
125	294	2	.3750	.5625	.0000		.0000
126	296	2	.5000	.5625	.0000		.0000
127	298	2	.6250	.5625	.0000		.0000
128	300	2	.7500	.5625	.0000		.0000
129	302	2	.8750	.5625	.0000		.0000
130	304	2	1.0000	.5625	.0000		.0000
131	306	3	.0000	.6250	.0000	.0000	.0000
132	309	2	.0625	.6250	.0000		.0000
133	311	3	.1250	.6250	.0000	.0000	.0000
134	314	2	.1875	.6250	.0000		.0000
135	316	3	.2500	.6250	.0000	.0000	.0000
136	319	2	.3125	.6250	.0000		.0000
137	321	3	.3750	.6250	.0000	.0000	.0000
138	324	2	.4375	.6250	.0000		.0000
139	326	3	.5000	.6250	.0000	.0000	.0000
140	329	2	.5625	.6250	.0000		.0000
141	331	3	.6250	.6250	.0000	.0000	.0000
142	334	2	.6875	.6250	.0000		.0000
143	336	3	.7500	.6250	.0000	.0000	.0000
144	339	2	.8125	.6250	.0000		.0000
145	341	3	.8750	.6250	.0000	.0000	.0000
146	344	2	.9375	.6250	.0000		.0000
147	346	3	1.0000	.6250	.0000	.0000	.0000
148	349	2	.0000	.6875	.0000		.0000
149	351	2	.1250	.6875	.0000		.0000
150	353	2	.2500	.6875	.0000		.0000
151	355	2	.3750	.6875	.0000		.0000
152	357	2	.5000	.6875	.0000		.0000
153	359	2	.6250	.6875	.0000		.0000
154	361	2	.7500	.6875	.0000		.0000
155	363	2	.8750	.6875	.0000		.0000
156	365	2	1.0000	.6875	.0000		.0000
157	367	3	.0000	.7500	.0000	.0000	.0000
158	370	2	.0625	.7500	.0000		.0000
159	372	3	.1250	.7500	.0000	.0000	.0000
160	375	2	.1875	.7500	.0000		.0000
161	377	3	.2500	.7500	.0000	.0000	.0000
162	380	2	.3125	.7500	.0000		.0000
163	382	3	.3750	.7500	.0000	.0000	.0000
164	385	2	.4375	.7500	.0000		.0000
165	387	3	.5000	.7500	.0000	.0000	.0000
166	390	2	.5625	.7500	.0000		.0000
167	392	3	.6250	.7500	.0000	.0000	.0000
168	395	2	.6875	.7500	.0000		.0000

169	397	3	.7500	.7500	.0000	.0000	.0000
170	400	2	.8125	.7500	.0000		.0000
171	402	3	.8750	.7500	.0000	.0000	.0000
172	405	2	.9375	.7500	.0000		.0000
173	407	3	1.0000	.7500	.0000	.0000	.0000
174	410	2	.0000	.8125	.0000		.0000
175	412	2	.1250	.8125	.0000		.0000
176	414	2	.2500	.8125	.0000		.0000
177	416	2	.3750	.8125	.0000		.0000
178	418	2	.5000	.8125	.0000		.0000
179	420	2	.6250	.8125	.0000		.0000
180	422	2	.7500	.8125	.0000		.0000
181	424	2	.8750	.8125	.0000		.0000
182	426	2	1.0000	.8125	.0000		.0000
183	428	3	.0000	.8750	.0000	.0000	.0000
184	431	2	.0625	.8750	.0000		.0000
185	433	3	.1250	.8750	.0000	.0000	.0000
186	436	2	.1875	.8750	.0000		.0000
187	438	3	.2500	.8750	.0000	.0000	.0000
188	441	2	.3125	.8750	.0000		.0000
189	443	3	.3750	.8750	.0000	.0000	.0000
190	446	2	.4375	.8750	.0000		.0000
191	448	3	.5000	.8750	.0000	.0000	.0000
192	451	2	.5625	.8750	.0000		.0000
193	453	3	.6250	.8750	.0000	.0000	.0000
194	456	2	.6875	.8750	.0000		.0000
195	458	3	.7500	.8750	.0000	.0000	.0000
196	461	2	.8125	.8750	.0000		.0000
197	463	3	.8750	.8750	.0000	.0000	.0000
198	466	2	.9375	.8750	.0000		.0000
199	468	3	1.0000	.8750	.0000	.0000	.0000
200	471	2	.0000	.9375	.0000		.0000
201	473	2	.1250	.9375	.0000		.0000
202	475	2	.2500	.9375	.0000		.0000
203	477	2	.3750	.9375	.0000		.0000
204	479	2	.5000	.9375	.0000		.0000
205	481	2	.6250	.9375	.0000		.0000
206	483	2	.7500	.9375	.0000		.0000
207	485	2	.8750	.9375	.0000		.0000
208	487	2	1.0000	.9375	.0000		.0000
209	489	3	.0000	1.0000	.0000	.0000	.0000
210	492	2	.0625	1.0000	.0000		.0000
211	494	3	.1250	1.0000	.0000	.0000	.0000
212	497	2	.1875	1.0000	.0000		.0000
213	499	3	.2500	1.0000	.0000	.0000	.0000
214	502	2	.3125	1.0000	.0000		.0000
215	504	3	.3750	1.0000	.0000	.0000	.0000
216	507	2	.4375	1.0000	.0000		.0000
217	509	3	.5000	1.0000	.0000	.0000	.0000
218	512	2	.5625	1.0000	.0000		.0000
219	514	3	.6250	1.0000	.0000	.0000	.0000
220	517	2	.6875	1.0000	.0000		.0000
221	519	3	.7500	1.0000	.0000	.0000	.0000
222	522	2	.8125	1.0000	.0000		.0000
223	524	3	.8750	1.0000	.0000	.0000	.0000
224	527	2	.9375	1.0000	.0000		.0000
225	529	3	1.0000	1.0000	.0000	.0000	.0000

BOUNDARY CONDITIONS
****

NODE	U-FIXED	PRESSURE	V-FIXED
209			4.0000
209	5.0000		
210			4.0000

210	5.0000	
211		4.0000
211	5.0000	
212		4.0000
212	5.0000	
213		4.0000
213	5.0000	
214		4.0000
214	5.0000	
215		4.0000
215	5.0000	
216		4.0000
216	5.0000	
217		4.0000
217	5.0000	
218		4.0000
218	5.0000	
219		4.0000
219	5.0000	
220		4.0000
220	5.0000	
221		4.0000
221	5.0000	
222		4.0000
222	5.0000	
223		4.0000
223	5.0000	
224		4.0000
224	5.0000	
225		4.0000
225	5.0000	
1		.0000
2		.0000
3		.0000
4		.0000
5		.0000
6		.0000
7		.0000
8		.0000
9		.0000
10		.0000
11		.0000
12		.0000
13		.0000
14		.0000
15		.0000
16		.0000
17		.0000
1	2.0000	
18	2.0000	
27	2.0000	
44	2.0000	
53	2.0000	
70	2.0000	
79	2.0000	
96	2.0000	
105	2.0000	
122	2.0000	
131	2.0000	
148	2.0000	
157	2.0000	
174	2.0000	
183	2.0000	
200	2.0000	
209	2.0000	
17	-2.0000	

```
  26  -2.0000
  43  -2.0000
  52  -2.0000
  69  -2.0000
  78  -2.0000
  95  -2.0000
 104  -2.0000
 121  -2.0000
 130  -2.0000
 147  -2.0000
 156  -2.0000
 173  -2.0000
 182  -2.0000
 199  -2.0000
 208  -2.0000
 225  -2.0000
   1              .0000
   VALT VALUES AT GAUSS-POINTS
1  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
2  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
3  1.0  1.0  1.0  4.0  1.0  1.0  4.0  1.0  1.0
4  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
5  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
6  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
7  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
8  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
9  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
10  4.0  4.0  4.0  4.0  1.0  1.0  1.0  1.0  1.0
11  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
12  4.0  1.0  1.0  4.0  4.0  1.0  4.0  4.0  1.0
13  1.0  1.0  1.0  4.0  1.0  1.0  4.0  1.0  1.0
14  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
15  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
16  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
17  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
18  4.0  4.0  4.0  4.0  4.0  4.0  4.0  1.0  1.0
19  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
20  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
21  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
22  4.0  1.0  1.0  4.0  4.0  1.0  4.0  4.0  1.0
23  1.0  1.0  1.0  4.0  1.0  1.0  4.0  1.0  1.0
24  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
25  4.0  4.0  4.0  4.0  1.0  1.0  1.0  1.0  1.0
26  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
27  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
28  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
29  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
30  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
31  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
32  1.0  1.0  1.0  4.0  4.0  1.0  4.0  4.0  1.0
33  1.0  4.0  4.0  1.0  4.0  4.0  1.0  1.0  1.0
34  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
35  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
36  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
37  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
38  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
39  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
40  1.0  1.0  1.0  1.0  1.0  4.0  4.0  4.0  4.0
41  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
42  1.0  1.0  4.0  1.0  1.0  4.0  1.0  1.0  1.0
43  1.0  4.0  4.0  1.0  4.0  4.0  1.0  1.0  4.0
44  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
45  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
46  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
47  1.0  1.0  4.0  4.0  4.0  4.0  4.0  4.0  4.0
48  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
```

49	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
50	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
51	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
52	1.0	1.0	4.0	1.0	1.0	4.0	1.0	1.0	1.0
53	1.0	4.0	4.0	1.0	4.0	4.0	1.0	1.0	4.0
54	4.0	4.0	4.0	4.0	4.0	4.0	4.0	4.0	4.0
55	1.0	1.0	1.0	1.0	1.0	4.0	4.0	4.0	4.0
56	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
57	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
58	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
59	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
60	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
61	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
62	1.0	1.0	4.0	1.0	1.0	4.0	1.0	1.0	1.0
63	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
64	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

RESULTS FOR ITERATION NUMBER 1

NODE	U-VELOCITY	PRESSURE	V-VELOCITY
1	.200E+01	.000E+00	.000E+00
2	.174E+01		.000E+00
3	.147E+01	-.176E+03	.000E+00
4	.121E+01		.000E+00
5	.695E+00	-.129E+04	.000E+00
6	.217E+00		.000E+00
7	-.245E+00	-.118E+04	.000E+00
8	-.671E+00		.000E+00
9	-.972E+00	-.241E+03	.000E+00
10	-.123E+01		.000E+00
11	-.145E+01	.596E+02	.000E+00
12	-.164E+01		.000E+00
13	-.178E+01	.284E+03	.000E+00
14	-.187E+01		.000E+00
15	-.194E+01	.484E+03	.000E+00
16	-.197E+01		.000E+00
17	-.200E+01	.565E+03	.000E+00
18	.200E+01		.281E+00
19	.146E+01		.262E+00
20	.000E+00		.000E+00
21	-.168E+00		.418E+00
22	-.925E+00		.294E+00
23	-.143E+01		.214E+00
24	-.176E+01		.113E+00
25	-.193E+01		.490E-01
26	-.200E+01		.267E-01
27	.200E+01	.250E+02	.574E+00
28	.165E+01		.578E+00
29	.136E+01	.148E+03	.630E+00
30	.867E+00		.784E+00
31	.450E+00	-.531E+03	.872E+00
32	.223E+00		.861E+00
33	-.294E-01	-.589E+03	.835E+00
34	-.423E+00		.707E+00
35	-.791E+00	-.439E+03	.552E+00
36	-.108E+01		.493E+00
37	-.134E+01	-.201E+02	.407E+00
38	-.155E+01		.326E+00
39	-.171E+01	.423E+03	.259E+00
40	-.182E+01		.180E+00
41	-.191E+01	.571E+03	.116E+00
42	-.196E+01		.704E-01
43	-.200E+01	.605E+03	.526E-01
44	.200E+01		.906E+00
45	.121E+01		.107E+01

46	.456E+00		.117E+01
47	-.881E-01		.106E+01
48	-.605E+00		.943E+00
49	-.116E+01		.672E+00
50	-.162E+01		.405E+00
51	-.188E+01		.192E+00
52	-.200E+01		.896E-01
53	.200E+01	.427E+03	.135E+01
54	.158E+01		.137E+01
55	.111E+01	.645E+03	.147E+01
56	.726E+00		.152E+01
57	.456E+00	-.770E+03	.145E+01
58	.132E+00		.138E+01
59	-.134E+00	.339E+03	.130E+01
60	-.368E+00		.124E+01
61	-.608E+00	.640E+03	.116E+01
62	-.806E+00		.108E+01
63	-.990E+00	.412E+03	.100E+01
64	-.127E+01		.782E+00
65	-.151E+01	.382E+03	.539E+00
66	-.170E+01		.398E+00
67	-.183E+01	.658E+03	.270E+00
68	-.194E+01		.181E+00
69	-.200E+01	.878E+03	.149E+00
70	.200E+01		.175E+01
71	.107E+01		.193E+01
72	.449E+00		.175E+01
73	-.144E+00		.158E+01
74	-.625E+00		.138E+01
75	-.103E+01		.114E+01
76	-.140E+01		.823E+00
77	-.175E+01		.426E+00
78	-.200E+01		.223E+00
79	.200E+01	.111E+04	.214E+01
80	.153E+01		.229E+01
81	.107E+01	-.728E+02	.233E+01
82	.779E+00		.219E+01
83	.458E+00	-.360E+03	.206E+01
84	.147E+00		.195E+01
85	-.134E+00	.114E+03	.185E+01
86	-.395E+00		.174E+01
87	-.639E+00	.502E+03	.161E+01
88	-.868E+00		.147E+01
89	-.107E+01	.125E+04	.131E+01
90	-.125E+01		.115E+01
91	-.143E+01	.151E+04	.973E+00
92	-.157E+01		.807E+00
93	-.170E+01	.118E+04	.657E+00
94	-.187E+01		.407E+00
95	-.200E+01	.120E+04	.282E+00
96	.200E+01		.273E+01
97	.114E+01		.265E+01
98	.481E+00		.237E+01
99	-.116E+00		.212E+01
100	-.635E+00		.185E+01
101	-.109E+01		.154E+01
102	-.146E+01		.114E+01
103	-.176E+01		.726E+00
104	-.200E+01		.440E+00
105	.200E+01	.232E+04	.329E+01
106	.151E+01		.320E+01
107	.119E+01	.135E+03	.304E+01
108	.809E+00		.287E+01
109	.482E+00	.557E+00	.269E+01
110	.190E+00		.254E+01
111	-.905E-01	.671E+01	.239E+01

112	-.555E+00		.225E+01
113	-.603E+00	.472E+03	.210E+01
114	-.845E+00		.193E+01
115	-.108E+01	.907E+03	.175E+01
116	-.130E+01		.155E+01
117	-.150E+01	.162E+04	.132E+01
118	-.168E+01		.107E+01
119	-.183E+01	.235E+04	.832E+00
120	-.190E+01		.655E+00
121	-.200E+01	.299E+04	.500E+00
122	.200E+01		.376E+01
123	.111E+01		.336E+01
124	.465E+00		.300E+01
125	-.567E-01		.265E+01
126	-.548E+00		.234E+01
127	-.103E+01		.198E+01
128	-.151E+01		.152E+01
129	-.186E+01		.992E+00
130	-.200E+01		.549E+00
131	.200E+01	-.131E+04	.435E+01
132	.150E+01		.403E+01
133	.997E+00	-.136E+04	.362E+01
134	.694E+00		.343E+01
135	.442E+00	-.466E+03	.325E+01
136	.199E+00		.306E+01
137	-.107E-01	-.438E+02	.289E+01
138	-.224E+00		.273E+01
139	-.456E+00	.145E+03	.258E+01
140	-.691E+00		.242E+01
141	-.939E+00	.878E+03	.224E+01
142	-.121E+01		.201E+01
143	-.147E+01	.165E+04	.175E+01
144	-.171E+01		.145E+01
145	-.188E+01	.289E+04	.112E+01
146	-.205E+01		.781E+00
147	-.200E+01	.455E+04	.566E+00
148	.200E+01		.473E+01
149	.111E+01		.419E+01
150	.459E+00		.346E+01
151	.449E-01		.308E+01
152	-.305E+00		.280E+01
153	-.775E+00		.250E+01
154	-.139E+01		.203E+01
155	-.190E+01		.126E+01
156	-.200E+01		.597E+00
157	.200E+01	-.252E+04	.507E+01
158	.174E+01		.494E+01
159	.130E+01	-.231E+04	.461E+01
160	.101E+01		.421E+01
161	.814E+00	-.186E+04	.386E+01
162	.484E+00		.349E+01
163	.174E+00	-.850E+03	.321E+01
164	.447E-01		.309E+01
165	-.984E-01	-.296E+03	.298E+01
166	-.286E+00		.287E+01
167	-.502E+00	.245E+03	.276E+01
168	-.781E+00		.262E+01
169	-.118E+01	.161E+04	.237E+01
170	-.162E+01		.206E+01
171	-.194E+01	.527E+04	.152E+01
172	-.178E+01		.863E+00
173	-.200E+01	.515E+04	.617E+00
174	.200E+01		.537E+01
175	.167E+01		.478E+01
176	.136E+01		.398E+01
177	.979E+00		.355E+01

178	.257E+00		.314E+01
179	-.117E+00		.300E+01
180	-.751E+00		.282E+01
181	-.184E+01		.178E+01
182	-.200E+01		.535E+00
183	.200E+01	-.316E+04	.603E+01
184	.153E+01		.541E+01
185	.226E+01	-.373E+04	.440E+01
186	.242E+01		.408E+01
187	.223E+01	-.131E+04	.416E+01
188	.223E+01		.405E+01
189	.215E+01	-.150E+04	.379E+01
190	.183E+01		.358E+01
191	.156E+01	-.754E+03	.354E+01
192	.113E+01		.349E+01
193	.516E+00	-.872E+02	.332E+01
194	.210E+00		.311E+01
195	-.832E-01	-.748E+03	.306E+01
196	-.300E+00		.347E+01
197	-.108E+01	.679E+04	.309E+01
198	-.300E+01		.713E+00
199	-.200E+01	.553E+04	-.822E+00
200	.200E+01		.621E+01
201	.267E+01		.384E+01
202	.382E+01		.425E+01
203	.327E+01		.390E+01
204	.345E+01		.380E+01
205	.207E+01		.392E+01
206	.194E+01		.322E+01
207	-.415E+00		.463E+01
208	-.200E+01		-.127E+01
209	.200E+01	-.183E+05	.400E+01
210	.500E+01		.400E+01
211	.500E+01	.386E+04	.400E+01
212	.500E+01		.400E+01
213	.500E+01	-.461E+04	.400E+01
214	.500E+01		.400E+01
215	.500E+01	-.101E+04	.400E+01
216	.500E+01		.400E+01
217	.500E+01	-.498E+01	.400E+01
218	.500E+01		.400E+01
219	.500E+01	-.519E+04	.400E+01
220	.500E+01		.400E+01
221	.500E+01	.669E+04	.400E+01
222	.500E+01		.400E+01
223	.500E+01	-.138E+05	.400E+01
224	.500E+01		.400E+01
225	-.200E+01	.429E+05	.400E+01

LARGEST CHANGE OCCURS AT DOF NUMBER 531 CHANGE = 1.0000

RESULTS FOR ITERATION NUMBER 24

NODE	U-VELOCITY	PRESSURE	V-VELOCITY
1	.200E+01	.000E+00	.000E+00
2	.187E+01		.000E+00
3	.176E+01	-.265E+02	.000E+00
4	.166E+01		.000E+00
5	.108E+01	-.144E+03	.000E+00
6	.691E+00		.000E+00
7	.264E+00	-.193E+03	.000E+00
8	-.283E+00		.000E+00
9	-.600E+00	-.114E+03	.000E+00
10	-.824E+00		.000E+00
11	-.103E+01	-.889E+02	.000E+00
12	-.128E+01		.000E+00
13	-.146E+01	-.752E+02	.000E+00
14	-.164E+01		.000E+00
15	-.179E+01	-.732E+02	.000E+00
16	-.189E+01		.000E+00
17	-.200E+01	-.844E+02	.000E+00
18	.200E+01		.143E+00
19	.000E+00		.000E+00
20	.943E+00		.464E+00
21	.278E+00		.442E+00
22	-.564E+00		.297E+00
23	-.102E+01		.235E+00
24	-.145E+01		.170E+00
25	-.179E+01		.120E+00
26	-.200E+01		.110E+00
27	.200E+01	-.157E+02	.287E+00
28	.177E+01		.306E+00
29	.162E+01	.193E+02	.380E+00
30	.108E+01		.710E+00
31	.677E+00	-.740E+02	.880E+00
32	.508E+00		.881E+00
33	.329E+00	-.134E+03	.867E+00
34	-.395E-01		.699E+00
35	-.431E+00	-.161E+03	.487E+00
36	-.702E+00		.468E+00
37	-.976E+00	-.120E+03	.447E+00
38	-.124E+01		.425E+00
39	-.142E+01	-.575E+02	.413E+00
40	-.159E+01		.349E+00
41	-.176E+01	-.567E+02	.278E+00
42	-.188E+01		.236E+00
43	-.200E+01	-.771E+02	.219E+00
44	.200E+01		.511E+00
45	.134E+01		.826E+00
46	.633E+00		.107E+01
47	.229E+00		.101E+01
48	-.256E+00		.947E+00
49	-.799E+00		.743E+00
50	-.137E+01		.563E+00
51	-.173E+01		.416E+00
52	-.200E+01		.341E+00
53	.200E+01	.404E+02	.913E+00

54	.162E+01		.954E+00
55	.117E+01	.732E+02	.121E+01
56	.829E+00		.133E+01
57	.622E+00	-.930E+02	.130E+01
58	.358E+00		.125E+01
59	.138E+00	.264E+02	.121E+01
60	-.666E-01		.118E+01
61	-.289E+00	.218E+02	.115E+01
62	-.477E+00		.114E+01
63	-.637E+00	-.605E+02	.112E+01
64	-.950E+00		.926E+00
65	-.125E+01	-.937E+02	.697E+00
66	-.148E+01		.628E+00
67	-.169E+01	-.639E+02	.545E+00
68	-.187E+01		.480E+00
69	-.200E+01	-.345E+02	.472E+00
70	.200E+01		.125E+01
71	.109E+01		.164E+01
72	.590E+00		.154E+01
73	.821E-01		.146E+01
74	-.343E+00		.134E+01

75	-.714E+00		.122E+01
76	-.111E+01		.109E+01
77	-.158E+01		.793E+00
78	-.200E+01		.605E+00
79	.200E+01	.133E+03	.160E+01
80	.152E+01		.186E+01
81	.106E+01	.873E+01	.198E+01
82	.838E+00		.188E+01
83	.566E+00	-.559E+02	.181E+01
84	.292E+00		.175E+01
85	.524E-01	-.298E+02	.169E+01
86	-.176E+00		.163E+01
87	-.396E+00	-.320E+01	.155E+01
88	-.603E+00		.147E+01
89	-.793E+00	.985E+02	.139E+01
90	-.967E+00		.132E+01
91	-.115E+01	.102E+03	.125E+01
92	-.131E+01		.120E+01
93	-.147E+01	-.878E+01	.115E+01
94	-.177E+01		.923E+00
95	-.200E+01	.141E+02	.745E+00
96	.200E+01		.228E+01
97	.112E+01		.227E+01
98	.550E+00		.208E+01
99	.278E-01		.192E+01
100	-.429E+00		.177E+01
101	-.839E+00		.161E+01
102	-.120E+01		.142E+01
103	-.155E+01		.125E+01
104	-.200E+01		.106E+01
105	.200E+01	.265E+03	.295E+01
106	.145E+01		.280E+01
107	.116E+01	.337E+02	.265E+01
108	.800E+00		.250E+01
109	.509E+00	-.174E+02	.235E+01
110	.254E+00		.224E+01
111	.108E-01	-.592E+02	.215E+01
112	-.218E+00		.207E+01
113	-.436E+00	-.116E+02	.199E+01
114	-.651E+00		.190E+01
115	-.857E+00	.282E+02	.181E+01
116	-.106E+01		.170E+01
117	-.125E+01	.957E+02	.160E+01
118	-.143E+01		.150E+01
119	-.162E+01	.168E+03	.140E+01

120	-.176E+01		.133E+01
121	-.200E+01	.350E+03	.127E+01
122	.200E+01		.343E+01
123	.104E+01		.294E+01
124	.454E+00		.263E+01
125	-.276E-02		.237E+01
126	-.421E+00		.221E+01
127	-.853E+00		.201E+01
128	-.127E+01		.178E+01
129	-.162E+01		.161E+01
130	-.200E+01		.152E+01
131	.200E+01	-.118E+03	.405E+01
132	.153E+01		.367E+01
133	.932E+00	-.126E+03	.314E+01
134	.643E+00		.297E+01
135	.397E+00	-.305E+02	.285E+01
136	.166E+00		.271E+01
137	-.213E-01	.210E+02	.261E+01
138	-.199E+00		.251E+01
139	-.395E+00	-.419E+02	.241E+01
140	-.597E+00		.232E+01

141	-.809E+00	.470E+01	.223E+01
142	-.104E+01		.211E+01
143	-.126E+01	.107E+03	.198E+01
144	-.145E+01		.185E+01
145	-.160E+01	.936E+02	.175E+01
146	-.185E+01		.176E+01
147	-.200E+01	.319E+03	.180E+01
148	.200E+01		.430E+01
149	.126E+01		.385E+01
150	.409E+00		.306E+01
151	-.555E-01		.276E+01
152	-.363E+00		.259E+01
153	-.729E+00		.244E+01
154	-.122E+01		.223E+01
155	-.163E+01		.190E+01
156	-.200E+01		.197E+01
157	.200E+01	-.162E+03	.448E+01
158	.193E+01		.442E+01
159	.164E+01	-.243E+03	.427E+01
160	.131E+01		.404E+01
161	.926E+00	-.212E+03	.373E+01
162	.383E+00		.323E+01
163	-.808E-01	-.131E+03	.283E+01
164	-.203E+00		.277E+01
165	-.329E+00	-.281E+02	.273E+01
166	-.475E+00		.269E+01
167	-.633E+00	.756E+01	.265E+01
168	-.816E+00		.260E+01
169	-.112E+01	-.291E+02	.248E+01
170	-.146E+01		.234E+01
171	-.171E+01	.350E+03	.220E+01
172	-.173E+01		.209E+01
173	-.200E+01	.113E+03	.203E+01
174	.200E+01		.460E+01
175	.180E+01		.432E+01
176	.146E+01		.398E+01
177	.623E+00		.337E+01
178	-.173E+00		.288E+01
179	-.556E+00		.281E+01
180	-.893E+00		.276E+01
181	-.166E+01		.262E+01
182	-.200E+01		.206E+01
183	.200E+01	-.326E+03	.479E+01
184	.167E+01		.466E+01
185	.200E+01	-.277E+03	.430E+01

186	.204E+01		.415E+01
187	.184E+01	-.205E+03	.408E+01
188	.188E+01		.400E+01
189	.181E+01	-.208E+03	.381E+01
190	.127E+01		.354E+01
191	.834E+00	-.126E+03	.343E+01
192	.240E+00		.321E+01
193	-.437E+00	-.106E+03	.298E+01
194	-.594E+00		.291E+01
195	-.660E+00	-.219E+02	.294E+01
196	-.774E+00		.317E+01
197	-.135E+01	.126E+03	.322E+01
198	-.288E+01		.212E+01
199	-.200E+01	.242E+03	.171E+01
200	.200E+01		.530E+01
201	.195E+01		.387E+01
202	.311E+01		.421E+01
203	.251E+01		.391E+01
204	.272E+01		.378E+01
205	.999E+00		.377E+01
206	.406E+00		.309E+01

207	-.217E+01		.484E+01
208	-.200E+01		-.178E-01
209	.200E+01	-.117E+04	.400E+01
210	.400E+01		.400E+01
211	.400E+01	.190E+03	.400E+01
212	.400E+01		.400E+01
213	.400E+01	-.459E+03	.400E+01
214	.400E+01		.400E+01
215	.400E+01	-.194E+03	.400E+01
216	.400E+01		.400E+01
217	.400E+01	-.875E+02	.400E+01
218	.400E+01		.400E+01
219	.400E+01	-.394E+03	.400E+01
220	.400E+01		.400E+01
221	.400E+01	.162E+03	.400E+01
222	.400E+01		.400E+01
223	.400E+01	-.499E+03	.400E+01
224	.400E+01		.400E+01
225	-.200E+01	.122E+04	.400E+01

LARGEST CHANGE OCCURS AT DOF NUMBER 332 CHANGE = 00.01

ME-1991-M-GHO-MOD